AD-A229 617

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

90 12 26 098

2

AFIT/GSM/LSY/90S-5

DEVELOPMENT OF A HANDBOOK FOR CONFIGURATION MANAGERS WITH APPLICATIONS FOR THE PROGRAM/SYSTEM IN FULL-SCALE DEVELOPMENT

THESIS

James E. Corbin, Captain, USAF

AFIT/GSM/LSY/90S-5

Approved for public release; distribution unlimited

The opinions and conclusions in this paper are those of the author and are not intended to represent the official position of the DOD, USAF, or any other government agency.

AFIT/GSM/LSY/90S-5

DEVELOPMENT OF A HANDBOOK FOR CONFIGURATION MANAGERS WITH APPLICATIONS FOR THE PROGRAM/SYSTEM IN FULL-SCALE DEVELOPMENT

THESIS

Presented to the Faculty of the School of Systems and Logistics

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Systems Management

James E. Corbin, B.S.

Captain, USAF

September 1990

Approved for public release; distribution unlimited

Preface

The purpose of this study was to develop a handbook that could be used to provide initial training on the discipline of configuration management to configuration managers and other program office personnel. In addition, since the applicability of each of the four individual configuration management processes for a program office is dependent upon which stage of the system acquisition life cycle the program is currently in, this thesis began the process of identifying suggestions to help the configuration managers apply configuration management to a program in the full-scale development phase. Future research could be done to provide the same type of information for the other phases of the system acquisition life cycle.

In writing this thesis I have received a great deal of assistance from others. I want to thank my faculty advisor, Mr. William Dean, for his diligence in reading, and providing comments on, the contents of the handbook. I would also like to thank: Ms. Peggy Jones, Ms. Debbie Martin, Ms. Doris Rebolt, Ms. Beverly Warren, Ms. Patty Wolpert, Mr. Ron Anthony, Mr. James Haynes, Lt Col Steve Kuprel, and Lt Dave Suh; who as configuration managers at Aeronautical Systems Division took time out of their busy schedules to read, and provide comments on, the handbook.

Finally, I wish to thank my family (Sandy, Christopher, and Milissa) for their understanding and patience when their husband, and father, was in "that mood."

James E. Corbin

Table of Contents

List of Tables

Abstract

This study resulted in the development of a Configuration Manager's Handbook that is intended to assist Air Force program offices and configuration management personnel apply the principles of configuration management to a product under development. The handbook can be used as a training document for in-coming personnel to a program office. It begins by briefly discussing the system acquisition life cycle as the domain in which a program is developed and the role of systems engineering in the development and design of the product. Configuration management is then introduced as a technical management control system that complements the technical actions undertaken during the systems engineering process. The handbook then proceeds to introduce, and in subsequent sections describe, the four processes that comprise configuration management: configuration identification, configuration audits, change management, and configuration status accounting. Finally, the uses of these processes for the program in full-scale development are discussed. This includes providing suggestions for the contents of Operating Instructions that should be produced by configuration managers to describe the specific applications of configuration management principles within the program office, and of Statement of Work tasks that should be required of the contractor.

DEVELOPMENT OF A HANDBOOK FOR CONFIGURATION MANAGERS

WITH APPLICATIONS FOR THE PROGRAM/SYSTEM

IN FULL-SCALE DEVELOPMENT

I. Introduction

This thesis will provide a handbook that can be used as an initial in-house training guide on the role of configuration management in the acquisition of a system to individuals being assigned to the configuration management branch/directorate (and it can also be used to explain configuration management to program/project managers) of the program office. In addition, the handbook will present applications of configuration management for the program/system in the full-scale development phase of the system acquisition process.

General Issue

The recent developments associated with the possible reunification of Germany, the democratic reforms being undertaken in the Eastern Bloc communist countries, and the implementation of the acquisition changes resulting from the Defense Management Review will have long lasting effects on the Department of Defense. Two of the results from these developments which will have large impacts on the acquisition environment are the pressure of shrinking budgets and the shortages of qualified personnel as the force structure is reduced. However, even with these impacts, program/acquisition managers will remain under continuous pressure to provide systems that meet the

using commands' increasingly sophisticated performance requirements, that are logistically supportable, and that are developed within cost and schedule constraints. One of the disciplines that the program manager can use to help achieve success in an acquisition program is configuration management.

During the system acquisition process, program managers can use configuration management as a technical management control system over the technical actions undertaken during system design. The configuration management process allows the program manager to measure achieved performance versus required performance; and to ensure that adequate documentation is maintained to monitor and control the identity of the developed system for future reference. By correctly applying configuration management throughout the system's life cycle, the program manager is assured of the definition, and control, of the technical content of the contract (3:12). Through configuration management, the configuration of each operational unit is known, and the supporting command's ability to support these operational units is enhanced.

Although current acquisition strategy in Department of Defense (DOD) directives, and in various DOD, service, and governmental agency regulations, prescribes specific requirements (policies) with regards to the accomplishment of specific activities in the area of configuration management (3:10), there is currently no type of handbook, training aide, or quick-look reference guide, that can be used by the configuration managers in the program office to better assist them in performing their responsibilities. In fact, the rule-of-thumb has been for "new" members arriving in a program office to be assigned to the configuration management directorate and learn by way of "on-the-job-training." The result in many program offices has been inadequate training and the resulting ineffective application of configuration management to the developing systems. A common risk in the transition of a product

from its development stage to its production stage has been the failure of the program office to have established, applied, and maintained an effective configuration management process during full-scale development (12:19). Mr. Thomas Shaw, Engineering Vice President of Sanders Associates Inc., states that this lack of an effective application of configuration management:

> ...leads to many pitfalls including an unknown baseline, excess production rework, poor spares effort, stock purging rather than stock control, and an inability to resolve field problems, all of which contribute to increasing program costs and lengthening program schedules. (12:19)

Specific Problem

An effective configuration management system within an acquisition program office is the result of the combination of a complete, contractually established configuration management program performed by the contractor, and a complete, established configuration management program in the the program office. Currently, governmental configuration management employees are randomly assigned to the program office through some type of personnel office action (that is the military members are assigned by the program manager upon arrival on permanent-change-of-station orders, and the civilian members are assigned to configuration management after being hired by the program office). Their initial contact with the discipline of configuration management is by a supervisor, or co-worker, providing some regulations, standards, and/or directives for them to read and "digest." These individuals are then provided (the timing to attend depending upon the program office's ability to obtain training spaces) Air Force Institute of Technology-offered short courses (SYS 026, Introduction to Configuration Management, and SYS 226, Applied Configuration Management) as their education base, along with the volumes of military documents.

After completing these courses, the configuration managers return to the system program offices where they are often called upon to establish a configuration

management program, including the contractual requirements to be levied on the contractor, with only minimal (if any) assistance/guidance from experienced configuration managers. Lacking this interface, and given the wait for the training space to be approved and provided for the employee, there should be some method of providing initial training to the fledgling configuration manager. Using that intimatating stack of military regulations and/or standards does not work well. Something else must be available to provide assistance and guidance to these configuration managers in setting up a configuration management program for the program office.

Thus, the research objective is to develop a training handbook which will provide them with both an overview, and many details, of the discipline of configuration management as it applies to system development. This will require that each of the processes that comprise configuration management be identified and discussed in some detail. In addition, for the handbook to be useful for configuration managers after their training has been started, it should also provide suggestions and insights to be used in the application of configuration management during the various stages of the system acquisition life cycle. Since the role of configuration management becomes increasingly important towards the end of the concept demonstration/validation phase and the start of the full-scale development phase of the life cycle, the handbook will include applications/requirements that can be levied on the contractor, and those that can be performed in the program office, to provide for a successful configuration management program for a major system during the full-scale development phase of the system acquisition cycle.

Investigative Questions

To accomplish this research objective, the four processes of configuration management [configuration identification, configuration audits and design reviews,

change management (to include configuration control and change control), and configuration status accounting] will be identified and assessed as to what constitute their essential activities which should be included as information in a training handbook. In addition, what specific activities/requirements associated with each of the processes should be required of the contractor, and/or performed by the configuration manager in the program office, to meet the configuration management requirements during full-scale development. The investigative questions of this research are therefore:

1. What should be included in the developed handbook that will (a) assist in the training of configuration managers such that they understand the configuration identification process of configuration management, and (b) assist configuration managers to determine how to apply configuration identification requirements (with respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

2. What should be included in the developed handbook that will (a) assist in the training of configuration managers such that they understand the role of design reviews in the systems engineering process and the role of configuration audits in the configuration management process, and (b) assist configuration managers to determine how to apply configuration audit requirements (with respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

3. What should be included in the developed handbook that will (a) assist in the training of configuration managers such that they understand the change management process (as it is applied to both the technical and non-technical portions of the program) of configuration management, and (b) assist configuration managers to determine how to apply change management requirements (with respect to those

expected of the contractor and those expected of the program office) for a product (and a program) in the full-scale development phase of the system acquisition life cycle?

4. What should be included in the developed handbook that will (a) assist in the training of configuration managers such that they understand the configuration status accounting process of configuration management, and (b) assist configuration managers to determine how to apply configuration status accounting requirements (with respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

Scope of the Research

In answering the above investigative questions, the research will begin by examining the role of configuration management in the system acquisition process and provide information on how the four processes are employed to assist the program office during system development. Due to the military applications of this research, the main sources of information will include military standards, regulation, pamphlets, and reports of joint Industry/Government configuration management workshops. Emphasis will be placed on the functions of configuration management which should be implemented to assure supportability of a system when it is in the Government inventory.

II. Literature Review

Method of Treatment and Organization

This literature review provides background information for the development of a handbook that (1) can be used to provide initial training on the discipline of configuration management to configuration managers and other program office personnel, and (2) provides suggestions and guidance to the application of the configuration management processes for the full-scale development phase of the system acquisition life cycle. First, the concept of configuration management (CM) and the factors which establish its need will be addressed. After the concept has been defined, the four processes of CM will be examined for their role in the acquisition of systems and system development.

Concept of Configuration Management

The system acquisition life cycle is the framework for the process through which the military services procure their weapon systems. To assure that the design decisions being made as the weapons system is under development address the impact on all elements of the system, and not just the immediate component being designed, the system design evolves through a process that is known as systems engineering (3:67). Configuration management is used to monitor this process, and to assist the program office in documenting the design qualified to meet the system performance requirements. Configuration management provides a technical management control system, complementing the actions of the systems engineering process, to ensure that the results are incorporated into the appropriate design documentation and that the documentation is then placed under Government control at

an appropriate point in the life cycle. Configuration management is a set of engineering practices that an organization can use to insure the integrity and continuity of the program information, engineering data, and design decisions that are made during the course of a project (13:54; 4:1). According to military standards and regulation, configuration management is:

> A discipline applying technical and administrative direction and surveillance to: (1) identify and document the functional and physical characteristics of a configuration item; (2) control changes to those characteristics; and (3) record and report change processing and implementation status. (9:2; 8:5)

The use of the CM processes requires careful selection of the documentation to be acquired and is implemented based on the size and complexity of the program, the stage of the acquisition life cycle, and the quantity of units of the item to be purchased (4:10). The CM processes assist the program manager in achieving system performance, a realistic schedule, and logistics supportability (6:2). In addition, CM allows for latitude in the design and development of the system at the same time that it introduces control at appropriate times in the acquisition process (6:2). Configuration managers perform this CM process through configuration identification, configuration audits and design reviews, change management (which includes aspects of both configuration control and change control), and configuration status accounting.

Configuration Identification

Through the use of configuration identification, technical documentation (including specifications, drawings, and parts lists) is selected that describes both the functional and physical characteristics of hardware and software components (2:B-5). Using the appropriate documents, the configuration identification is used to establish baselines which contractually define progressively more detailed descriptions of the items being

bought and which provide a basis for measuring the contractor's technical progress by comparing it to these contractual requirements. These baselines are established at points in a program when it is normally deemed necessary to establish a definable and manageable departure point for the development and production of the system (8:5). The baseline, plus any approved changes to it, constitutes the current contractually binding technical definition of what the Government expects the item/system to accomplish.

Baselines. Baseline management is an important aspect of the function of configuration identification. The requirements in the baseline comprise the contractual agreements about the technical basis for the system (2:B-4). As the contractor progresses with design and development, the baselines are used to provide more detailed definition of the system performance and design. Configuration management is usually concerned with establishing three baselines: functional, allocated, and product.

Functional Baseline. This is the first baseline established during the development and acquisition of a system. The functional baseline is used to formally document the performance, interfaces, operational requirements, design constraints, and logistics support constraints (7:10). At the highest level in the system, these functional requirements are usually contained in a single system specification that comprises the contractual technical definition of the system capabilities. This technical agreement provides the technical basis for the development process of the system (4:4). With the system level requirements agreed to, the program will then proceed to identify critical elements which are technically defined in the next baseline.

Allocated Baseline. Once the top-level system functional requirements have been generated, the contractor turns his attention toward the development of requirements for the major subsystems. The allocated baseline for each important subsystem (called a configuration item or CI) is used to more completely define and document the functional characteristics of that CI as a part of the overall system (4:4). The configuration identification documentation used to define these CI functional characteristics is either a development specification (for hardware CIs) or a requirements specification (for computer software CIs, CSCIs) (5:16). The development and requirements specifications define the functional characteristics, interfaces with other subsystems, establish any design constraints (e.g., logisitics, personnel, security) that pertain to the component, and include the tests required to show compliance of each of the CI/CSCIs to their performance characteristics (5:84). When established as the allocated baseline for a CI or CSCI, this technical contract provides the basis for the contractor to proceed into detailed design, development, building of test prototype, and testing of the CI or CSCI.

Product Baseline. After the contractor completes the design and testing of each CI/CSCI and successfully demonstrates that the CI/CSCI meets the specified technical requirements previously stated, the contractor provides the detail design documentation that is used to establish the product baseline (5:16). That documentation normally describes the exact physical design (in terms of drawings, computer source codes, and/or parts lists) of the CI/CSCI, the required performance characteristics that should be tested during production, and the acceptance tests to verify these characteristics (7:12). This last baseline establishes the technical criteria for the manufacture and subsequent acceptance of production units of the CI/CSCI.

Configuration Audits

The second of the major areas of configuration management is configuration audits. As the contractor completes the design and testing of the system and its CI/CSCIs, the Government checks each CI/CSCI for compliance with the requirements of its baselined configuration identification. The configuration audit function of CM verifies and validates the achievement of the performance requirements by the CI/CSCIs and the system and the complete and accurate documentation of its detailed design prior to control being turned over to the Government (2:B-9). To accomplish these efforts, both functional and physical configuration audits are performed. In some cases, where system level tests are required to verify system performance requirements have been met, the Government may also require the contractor to conduct a functional system audit(s) (formerly referred to as a formal qualification review) (6:12).

Functional Configuration Audit. At separate functional configuration audits each of the CI/CSCIs developed by the contractor or one of his vendors is examined to verify that the CI/CSCI has achieved the required performance specified in the allocated baseline (4:6). The audit will provide formal acknowledgement that the CI/CSCI design has met the requirements defined in the development or requirements specifications (5:71). The successful completion of the functional audit insures that the performance defined in the allocated baseline has been achieved, and that the CI/CSCI's configuration is ready to be released for production (5:72).

Functional System Audit. As the program undergoes development, the requirements associated with the system's functional baseline are allocated down to its CIs and CSCIs, and these requirements are used to establish and define each of the

CI/CSCI's allocated baseline. For most programs, once all the CI/CSCIs performances requirements have been verified through their functional configuration audits (FCAs), the system performance has also been verified (5:72). In cases where a very complex system is involved, a functional system audit (FSA) may be required to evaluate the results of the system level tests to verify that the performance requirements specified in the current approved System/System Segment Specification have been met (6:12). Even if thought to be appropriate for the program, unlike FCAs, the FSA is not a prerequisite for the program to conduct a physical configuration audit (PCA) (5:72). Wherever possible, however, the system level verifications should be accomplished as a part of the FCAs so that system-level problems are identified prior to the CI/CSCI's PCA (10:83).

Physical Configuration Audit. When production units (or operational copies of computer software) of the subsystem are ready to be delivered, a comparison of the actual deliveral item to the technical documentation must be performed. The physical configuration audit is performed to verify that the detailed design cited in the product specification matches the "as-built" unit configuration (4:7). The audit includes a detailed review of "engineering drawings, specifications, software listings and design documents, and other technical/manufacturing data used in producing the configuration item" (2:B-10). In addition to the design documentation, the contractor's engineering change release system is reviewed to verify that the contractor has the ability to control changes to the detailed design documentation (4:7). When the contractor has successfully completed the physical configuration audit, the product baseline is normally established and the Government takes control of the detail design.

Change Management

Once the contract (a new one at the start of each of the phases of the system acquisition life cycle) has been signed by both the contractor and the Government, and as each technical baseline is established, there must be procedures in place to regulate any changes proposed to the documentation that comprises either the program's technical or contractual baselines. The program office is able to control these baselines through the application of change management. Change management involves the evaluation, coordination, and decision (approval or rejection) of all proposed changes (2:5-3; 3:85). Change management is composed of two parts - configuration control and change control.

Configuration Control. The procedures used by the contractor, configuration managers, and other program office personnel to regulate the flow of proposed changes to the technical documentation constituting the system's functional baseline, or its CI/CSCI's allocated and/or product baselines are referred to as configuration control (4:1). According to Military Standard 480B, configuration control is:

> The systematic proposal, justification, evaluation, coordination, approval or disapproval of proposed changes, and the implementation of all approved changes in the configuration of a CI after formal establishment of its baseline. (7:6)

Thus, configuration control assures that the complete impact of a change to an established configuration is considered prior to an approval/disapproval decision being made. Additionally, configuration control insures that the proposed changes are beneficial. Will the change correct a deficiency, improve the operational capabilities or logistics supportability of the system, effect a cost savings, or beneficially affect approved delivery schedules (2:B-10)? Documented changes to the baseline take one of three forms: engineering change proposals (ECPs), deviations, or waivers.

Engineering Change Proposals. If a change should be made to requirements contained in baselined configuration identification, an ECP is submitted which provides sufficient information to document all impacts of the proposed change (4:7). As the configuration identification of the product evolves, the amount of information required in the ECP increases (because of the amount of information controlled by the baselines). For the functional baseline, an ECP may only describe specification wording changes plus the qualitative impacts of the change on the performance requirements or logistics support of the system (4:7). However, as the system matures and a product baseline is approved, the discussion of the impact of the change can include changes in part design, retrofit requirements, and specific impacts on the logistics supportability (e.g. spares, test software) of the system (2:B-10; 4:7).

Deviations or Waivers. When temporary relief (usually for a single deliverable unit of the CI/CSCI) from a technical requirement in an approved baseline is required because the existing requirement is deemed to be the preferable, the contractor submits either a deviation or waiver depending upon the assembly status of the unit(s) proposed to be affected (5:25). This relief allows a temporary departure from the approved design to a less-desirable design, but it is supposed to be the permanent configuration for the unit(s) involved. If the deficiency to the baseline is identified prior to the fabrication/manufacturing of the unit involved, then a deviation is submitted (2:5-3). A waiver is submitted when the deficiency is discovered while the unit involved is being assembled or has been submitted for acceptance (2:5-3).

Change Control. The procedures used by the contractor, configuration managers, and other program office personnel to regulate the flow of proposed changes to contractual requirements that do not impact the baselined technical requirements are

referred to as change control (2:5-3). The documents that are used for this aspect of change management are known as contract (task) change proposals (CCP/TCPs) (2:5-3). CCP/TCPs are primarily written against the requirements listed in the Statement of Work tasks, in contractually required delivered plans (e.g., test plans, System Engineering Management Plan), and contract data requirements lists (2:5-7). The program office must plan for reviewing and approving these CCP/TCPs in a manner similar to its established process for controlling ECPs (3:85).

Configuration Status Accounting

The first three aspects of configuration management establish the program baselines, validate the development effort and documentation of the resultant design, and provide a means to assure that the impacts of a proposed change are reviewed prior to its approval or disapproval. However, current and historical information relating to these three areas must be stored and available in a management information system. Configuration status accounting "provides traceability of the documentation, units, and activities resulting from the other three areas of CM" (4:8). The status accounting information system provides information required to identify the current configuration baseline and the status of both change proposals and implementation actions (9:iii); it may also track the priority of the changes, the schedule for completion of implementation activities, and the progress to date (5:30). Configuration status accounting also provides the means to track proposed changes from the time they are first submitted until they are approved or disapproved (4:8). If the change is approved, then status accounting provides the means to track all the proposed impacts of the change and the means to track and verify the implementation (2:B-14). Without configuration status accounting, program and/or configuration managers would not

have a way to monitor and track the implementation of, and the results obtained from, the other three qualities of configuration identification, auditing, and control.

Summary

From the literature reviewed, it can be concluded that maintaining a strong configuration management function is an important contribution of the technical viability of a program. Configuration management encompasses the engineering management practices and procedures used to oversee the functional and physical characteristics of a system as defined in the appropriate documentation. Using CM, the program baselines are identified, established, and verified to meet the requirements. CM also provides the structure for assessing proposed changes and for tracing all actions against the current baseline. In the hectic world of program acquisition, configuration management provides a way to establish the definition of, and maintain control of, the technical aspects of the things we buy.

III. Methodology

This chapter discusses alternative methods that could be used to accomplish the research objective of developing a useable handbook to provide initial training on the discipline of configuration management as it is applied to system development and to provide assistance and guidance in setting the configuration management program requirements for the program office as the product enters, and proceeds through, full-scale development.

The Military Documents Approach

The first method that could be used to accomplish the research objective would be to use available military standards and regulations to develop the handbook based on an interpretation of how to employ the requirements stated in these documents. This approach would be beneficial in that the handbook would be based on "official" Department of Defense (DOD) and Department of the Air Force (USAF) positions (stated policy and standard practices) on the role of configuration management in the acquisition of a major system. The following list of military documents (using the latest revisions available at the time of this writing) will be considered while determining the inputs required for the different sections of the handbook.

Document Number	Title
AFR 14-1	Configuration Management
AFP 57-1	Operational Needs, Requirements, and Concepts
AFR 800-2	Acquisition Management: Acquisition Program Management

AFR 800-14	Acquisition Management: Lifecycle Management of Computer Resources in Systems
AFSCP 800-3	A Guide for Program Management
AFSCP 800-7	Configuration Management
DOD-D-1000B	Drawings, Engineering and Associated Lists
DOD-STD-100C	Engineering Drawing Practices
DOD-STD-2167A	Defense System Software Development
MIL-N-7513F	Nomenclature Assignment, Contractors Method for Obtaining
MIL-S-83490	Specifications, Types and Forms
MIL-STD-480B	Configuration Control - Engineering Changes, Deviations, and Waivers
MIL-STD-481B	Configuration Control - Engineering Changes (Short Form), Deviations, and Waivers
MIL-STD-482A	Configuration Status Accounting Data Elements and Related Features
MIL-STD-483A (USAF)	Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs
MIL-STD-490A	Specification Practices
MIL-STD-1521B	Technical Reviews and Audits for Systems, Equipments, and Computer Programs
MIL-T-31000	Technical Data Packages, General Specification for

However, a disadvantage of this approach is that, since the handbook would be developed entirely from the military pamphlets, standards, and regulations, there would be no feedback as to its effectiveness within an actual program office environment. It

would be difficult to determine which portions of the military documents were not absolutely necessary in performing and maintaining the configuration management role in developing a product through the system acquisition cycle.

The Role Model Approach

Another possible method that could be used to develop the handbook would be to look for a program office that has been recognized as developing an "exceptional" configuration management (CM) program and using their developed CM program as a model of a quality CM system. The program office selected would be used as basis for determining what should be included in the training portion of the handbook and for determining what should be included in the tailoring of the configuration management processes for full-scale development.

But how can such an exemplary organization be selected? This "exceptional" CM organization could be found by reviewing Inspector General (IG) final reports and noting those program offices with high ratings for their configuration management organization. If more than one program was found to have a high rating for their CM organization, these could be compared to determine which phase of the system acquisition cycle they were in during their IG review. The one that was the furthest along in their full-scale development phase, or just entering the production phase, would be the program selected. [The rationale for this decision is that the handbook being developed will address those requirements of the CM processes that should be applied to those programs that are entering, or have already entered, the full-scale development phase of the system acquisition life cycle.] Once the final selection was made, that program office's CM shop would be used as the model after which all other configuration management organizations should be trained and the contractual CM

requirements would be used to provide suggestions as to how to apply CM requirements to the contractor and how to set up a CM program within the program office.

The handbook could include diagrams and description of their organizational structure; samples of the Statement of Work paragraphs, relating to configuration management, they used in their full-scale development contract; excerpts from the portion of their program management plans that related to configuration management; and copies of any Operating Instructions they developed that described how the configuration management process would be performed within their system program office.

However, the disadvantage of using this approach is that each program office is tasked with the development of its own unique product under its own set of ground rules and acquisition strategy. What works for one may not necessarily work for the development of another product. Each program office would still have to decide, with regards to developing their own specific product, how to utilize the model CM process to its advantage. Just copying the model organization, even though it was a successful configuration management program, might cause more problems than it solved, and it would not be tailored to the particular configuration and program managers' needs for the new program to help them produce an effective configuration management program for their product.

<u>The Survey Approach</u>

A third possible method that could be used to develop the proposed handbook would be to survey as many as possible, past and present, configuration and program managers on their attempts to establish a successful configuration management

program within the program office. Among other things they could be requested to provide information pertaining to how their CM organization was developed; what inputs they used to develop Operating Instructions that defined the processes that would be used to accomplish the configuration management function for full-scale development; what Statement of Work paragraphs they used to task the contractor with regards to configuration management; and what were the strong and weak points of their CM program used within the program office.

A disadvantage of this approach is that, by following this approach, the military documents that describe how configuration management should be considered by a program office might be ignored. There could be the possibility that the program offices that respond did not use the current documents now available when they set up their configuration management program within the program office. In fact, the program and configuration managers may not have had the appropriate expertise and may have received assistance of other programs without really determining if it was applicable for their specific product's development.

The Combination Approach

The final possible method that has been considered to accomplish the stated research objective would be a combination of two of the above mentioned alternatives. This approach would allow for the blending of the theoretical (The Military Documents Approach) and the actual (The Survey Approach) aspects to configuration management. Current military (pamphlets, standards, and regulations) and other technical documents could be reviewed to determine what the configuration/program manager should know about the configuration management discipline and what type of information is required to establish an effective CM program for the program office as

its product progresses through full-scale development. Information could also be obtained from current programs about their CM programs and how they are going about meeting the configuration requirements for their product. They could highlight the strong and weak points of their organization's established CM program as they see them. These inputs could then be compared to the requirements stated in the military documents and a comprehensive, yet plausible, training handbook could be developed that would be of some functional use for the newly assigned configuration or program manager.

Particular Method Selected

To best accomplish the research objective of developing a useable training handbook that will also provide assistance to the configuration manager apply the concepts of configuration management during full-scale development, the combination method will be followed. The implementation of this method will progress through three stages.

First Stage. The first stage will begin by looking at the system acquisition life cycle and the role of systems engineering within this cycle. Next, it will be determined how configuration management can be used to assist the program manager in managing the technical side of the program. This phase will be accomplished by performing an in-depth literature review of military documents (regulations, standards, pamphlets, and directives) pertaining to configuration management (CM), text material from Air Force Institute of Technology-offered short courses SYS 028 (Introduction to Configuration Management) and SYS 228 (Applied Configuration Management), reports of Industry/Government CM workshops, and Electronic Industries Association (EIA)-sponsored textbooks on the four individual processes of configuration management.

This stage will be used to determine what aspects of configuration identification, control, audits, and status accounting are applicable for a major system program or product (with tailoring emphasis on the full-scale development phase of the system acquisition cycle); how these processes should be used; and when they should be implemented during the acquisition cycle.

Military Documents. The military documents listed under The Military Documents Approach, based on their application to the role of configuration management in the development of a product as it progresses through the system acquisition life cycle, are candidates for review during the document/textbook review phase. The latest approved revision of each document, as of the time of this writing, will be the criteria for deciding which version to review.

Second Stage. After completing the literature review, the research will move into a review of current programs, including discussions with CM personnel. The purpose of this phase is to acquire some firsthand information about how current CM programs were set up for the full-scale development phase of the system acquisition life cycle. These reviews of current programs, and discussions with their configuration managers, will focus on the programs' Statement of Work taskings, contractor Configuration Management Plans, program office Operating Instructions on configuration management practices, and configuration management branch/directorate inputs into the Program Management Plans for the program. Programs selected for review will be those that included some sort of concept demonstration/validation phase activity prior to beginning full-scale development. Discussions will be conducted with program, product, and/or configuration managers selected based on their involvement with programs during full-scale development. Emphasis will be placed on determining the

effectiveness of the system's configuration management program, and in particular, the CM program's strong and weak points. The sources of information will be limited to several major programs (each of which is contained within its own program office) at Aeronautical Systems Division that meet the above acquisition cycle criteria for inputs. Candidate programs include Short Range Attack Missile (SRAM-II), F-15E, F-16, LANTIRN, B-1, and C-17.

At the conclusion of these program reviews and discussions, this phase of the research will continue with the writing of the configuration management handbook. This will involve analyzing the personal inputs from the interviewed managers to define the necessary CM processes that should be implemented to manage the program while it is in the full-scale development phase of the system acquisition life cycle. It also involves the comparison of those personal inputs with the policy/standard practices wherever possible to provide effective "real world" practices which illustrate, and provide options for implementation of, the policy.

Format. The format chosen for the handbook will be one that can first be used as a training aide to describe the overall role of configuration management as it pertains to the development of a product (in particular, one that is also considered to be a major program) through the system acquisition process. Next, the handbook will include a section describing how to apply the processes of configuration management during full-scale development to ensure that the technical management of the product development is adequate.

Third Stage. After the handbook is written, the final stage of the research will determine whether the developed handbook would be of any value for a program office to use in initially training their newly assigned configuration managers. This stage will

involve the distribution of the draft handbook for comments about the contents (in order to fine tune the wording) and for opinions about its utility for use as a training tool.

As the chapters for the handbook are developed, they will be provided to selected configuration managers currently assigned to program offices at Aeronautical Systems Division, who have agreed to review the handbook for its content and its utility. They will be requested to determine if the developed document could be used for training a newly assigned configuration manager on the principles of configuration management.

IV. Analysis and Findings

This chapter provides an analysis of how the particular method selected was used to develop a useable handbook that will (1) provide initial training on the discipline of configuration management as it is applied to system development, and (2) provide assistance and guidance in setting the configuration management (CM) program requirements for the program office as the product enters, and proceeds through, full-scale development.

First Stage of the Study

The initial stage of the particular method selected was used to answer the first part of each of the four investigative questions posed in Chapter I. In particular, the military documents (regulations, standards, pamphlets, and directives), Air Force Institute of Technology (AFIT)-offered short courses (SYS 028 and SYS 228) text materials, reports of Industry/Government CM workshops, and Electronic Industries Association-sponsored textbooks reviewed were used to answer the questions:

(1) What should be included in the developed handbook that will assist in the training of configuration managers such that they understand the configuration identification process of configuration management?

(2) What should be included in the developed handbook that will assist in the training of configuration managers such that they understand the role of design reviews in the systems engineering process and the role of configuration audits in the configuration management process?

(3) What should be included in the developed handbook that will assist in the training of configuration managers such that they understand the change management process (as it is applied to both the technical and non-technical portions of the program) of configuration management?

(4) What should be included in the developed handbook that will assist in the training of configuration managers such that they understand the configuration status accounting process of configuration management?

Table 1 (on page 28) shows which military documents and AFIT-offered short courses were reviewed, and used, in developing the training portion of the Configuration Manager's Handbook. The information included in each of these documents was compared to determine the different aspects associated with each of the CM processes. To ensure that a complete list of concerns were addressed, the emerging subjects were compared to an in-house training plan (that was proposed for use in industry) developed by a committee composed of Government and industry personnel at a workshop sponsored by the Government and the Electronic Industries Association (12:94-98). If inconsistencies were found between the documents, then the most current dated document was assumed to be the latest "official" position. The results of this stage of the study are included as Sections 3 through 8 of the developed Configuration Manager's Handbook, which is attached as Appendix HB.

Second Stage of the Study

The second stage of the study was performed to answer the second part of each of the posed investigative questions. That is, this stage was used to determine:

(1) What should be included in the developed handbook to assist configuration managers to determine how to apply configuration identification requirements (with

Table 1

Documents and Configuration Management Principles

	IDENTIFICATION	AUDITS	CHANGE MANAGEMENT	STATUS ACCOUNTING
MILITARY DOCUMENTS				
AFR 14-1	X	X	X	X
AFR 800-14	X	X	X	X
AFSCP 800-3	X	X	X	X
AFSCP 800-7	X	X	X	X
DOD-D-1000B	X			
DOD-STD-100C	X			
DOD-STD-2167A	X	X	X	X
MIL-N-7513F	X			
MIL-S-83490	X			
MIL-STD-480B			X	
MIL-STD-481B			X	
MIL-STD-482A				X
MIL-STD-483A	X		X	X
MIL-STD-490A	X		X	
MIL-STD-1521B		X		
MIL-T-31000	X			
AFIT COURSE TESTS				
SYS 028	X	X	X	X
SYS 228	X	X	X	X

respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

(2) What should be included in the developed handbook to assist configuration managers to determine how to apply configuration audit requirements (with respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

(3) What should be included in the developed handbook to assist configuration managers to determine how to apply change management requirements (with respect to those expected of the contractor and those expected of the program office) for a product (and a program) in the full-scale development phase of the system acquisition life cycle?

(4) What should be included in the developed handbook to assist configuration managers to determine how to apply configuration status accounting requirements (with respect to those expected of the contractor and those expected of the program office) for a product in the full-scale development phase of the system acquisition life cycle?

To accomplish this stage, current program offices at Aeronautical Systems Division (ASD) (located at Wright-Patterson Air Force Base, Ohio) were contacted for assistance in obtaining firsthand information as to how CM programs have been constructed for major systems in the past. The program offices contacted were: ASD/B1 (B-1 Program Office), ASD/VB (Tacit Rainbow Program Office), ASD/VF (F-15E Program Office), ASD/VL (LANTIRN Program Office), ASD/YC (C-17 Program Office), ASD/YG (SRAM-II Program Office), and ASD/YP (F-16 Program Office).

These were selected because each is a single product program office, and they are not considered "black-world" (classified) type programs.

To determine what to include in the handbook to assist configuration managers develop a CM program for the product entering full-scale development, a comparison was made of the Operating Instructions (those that pertained to CM) in effect in each of the selected program offices. Table 2 (shown on page 31) identifies which types of Operating Instructions were prepared and used by each program office. All the program office Operating Instructions (OIs) pertaining to configuration management were included in the analysis. However, each program office did not use the same name for the OIs, so the OIs were consolidated into groupings. For example, if the program office maintained an OI that discussed the change management process, then regardless of the title provided the OI by the program office it is shown under the listing for Change Management. Based on this input, Section 9.1 of the Configuration Manager's Handbook was prepared (see Appendix HB). This Section of the handbook includes suggested inputs that should be included in Operating Instructions, prepared by the program office's configuration management organization, that explain how each of the four principles of configuration management will be applied within the program office.

To determine what to include in the handbook to assist the configuration manager levy CM requirements on the contractor, the same program offices were asked for a listing of Statement of Work (SOW) tasking paragraphs currently included in their contracts. Table 3 (on page 32) reflects the separate CM tasks included in their contracts. [NOTE: Three of the program offices that provided OIs did not provide inputs of their full-scale development SOW paragraphs. The F-16 program office (ASD/YP) has been out of their initial full-scale development phase for such a long period of time that their original full-scale development contract has been stored and

Table 2

Operating Instructions Used in ASD Program Offices

	ASD/ B-1	ASD/ VB	ASD/ VF	ASD/ VL	ASD/ YC	ASD/ YG	ASD/ YP
SPECIFICATION MAINTENANCE	X	X				X	X
CONFIGURATION AUDITS	X		X	X		X	X
CONFIGURATION CONTROL BOARD (CCB)	X	X	X	X	X	X	X
PRE-CCB	X		X			X	
SOFTWARE COMPUTER CONFIGURATION SUB-BOARD							X
CHANGE MANAGEMENT	X	X	X	X	X	X	X

was not readily accessible. The SRAM-II program office (ASD/YG) requested that their SOW paragraphs not be used due to program sensitivities. The F-15E program office (ASD/YF) was not able to provide the SOW paragraphs because of schedule constraints.] The inputs obtained helped determine which work tasks are/should be required of a major system's contractor. The suggested SOW tasking paragraphs that a configuration manager could use for a product entering full-scale development are included as Section 9.2 of the Configuration Manager's Handbook (see Appendix HB).

Table 3

SOW Paragraphs Used in ASD Program Offices

	ASD/ B-1	ASD/ VB	ASD/ VF	ASD/ VL	ASD/ YC	ASD/ YG	ASD/ YP
BASELINE MANAGEMENT	X	X		X			
SPECIFICATION MAINTENANCE	X	X		X	X		
ENGINEERING DRAWINGS	X	X		X	X		
REQUESTS FOR NOMENCLATURE	X	X		X			
COMPUTER PROGRAM IDENT NUMBER	X	X		X	X		
CONFIGURATION AUDITS	X	X		X	X		
CHANGE MANAGEMENT	X	X		X	X		
CONFIGURATION STATUS ACCOUNTING	X	X		X	X		

Third Stage of the Study

Once the handbook was drafted it was submitted to experienced CM individuals assigned to ASD. These individuals were selected based on their association with major programs that were either still in, or had recently finished, the full-scale development phase of the system acquisition life cycle. The individuals selected also held different levels of responsibility within the program offices. This variation was used to determine if the handbook would have application to individuals with differing experience levels. In addition to individuals from seven different program offices, the

director of the configuration management division within the DCS for Integrated Engineering and Technical Management (to obtain ASD's comments) and a configuration manager from a multi-program program office (selected to determine if the first 8 sections of the handbook would be of assistance in providing training to CM personnel in other than major program offices) were also asked to review the handbook. The individuals who responded were:

(1) Mr James Haynes
Director of Configuration and Data Management
DCS Integrated Engineering and Technical Management

(2) Mr Ronald Anthony
Director, Configuration and Data Management Division
C-17 Program Office

(3) Ms Doris Rebolt
Chief, Configuration and Data Management Division
LANTIRN Program Office

(4) Ms Beverly Warren
Chief, Configuration and Data Management Division
SRAM-II Program Office

(5) Ms Peggy Jones
Chief, Configuration and Data Management Division
Training Systems Program Office

(6) Ms Debbie Martin
Director of Acquisition Support
Tacit Rainbow Program Office

(7) Lt Col Steve Kuprel
Deputy Director, Configuration Management
F-16 Program Office

(8) Lt Dave Suh
Configuration Manager
B-1 Program Office

(9) Ms Patty Wolpert
Configuration Management Specialist
F-15E Program Office

As each of the individuals finished reviewing the handbook, their specific comments on its content were discussed with them to determine whether the handbook should be expanded to include these comments or to determine if the comments were more specifically addressing tailoring aspects that were used within that particular program office. In addition, these discussions also involved their general comments about their perceptions as to the utility of the document. Their comments were analyzed by dividing them into three groups based on their job positions. The first group included Mr Haynes and the individuals within the DCS for Integrated Engineering and Technical Management (ASD/ENC) who work for Mr Haynes and who had also reviewed the handbook. The second group of individuals include those whose job title is either Director, Deputy Director, or Chief for their division (Configuration and Data Management or Acquisition Support). This group includes those identified in 2 through 7 on the list provided on page 33. The final group is comprised of the configuration manager/specialists at the working level. This group includes the last two individuals listed on page 33.

The first group of individuals (those in ASD/ENC) had a mixed response to the handbook. Given their desire to have a document developed that provides detailed directions on how to apply the principles of configuration management, their initial response was that the information provided in the handbook was too general and contained more basic information than they thought necessary (1:1). Their initial premise when reviewing the developed handbook was that the configuration manager would have completed the initial training that is provided at AFIT (1:2). Yet, this premise was in direct contradiction to that of the research project. When this was brought to their attention, their opinion was then changed to "...the information is

generally complete and accurate, [and] understandably void of program office "peculiarities" in policy and method" (11). Based on the objective that the developed handbook provide a viable initial training approach for configuration managers, they agreed that the handbook could be used by program offices as an instrument in introducing the principles of configuration management.

There were two prevalent responses from the second group of individuals who reviewed the handbook. The first response was a general consensus that, although Air Force policy as stated in AFR 14-1 requires that the allocated baseline be established at the System Design Review (or no later than completion of the Preliminary Design Review, PDR) (6:6,21), that in "reality" the allocated baseline will almost never be established prior to the PDR. For most programs, the contractor is only expected to provide a draft copy of the development and/or requirements specification(s) for the hardware and computer software configuration items (CI/CSCIs) at the System Design Review. It is more likely that these specifications will be available for authentication, and therefore used to establish the allocated baseline for the CI/CSCI, at the respective CI/CSCI's Preliminary Design Review (and in some instances even as late as the CI/CSCI's Critical Design Review). However, since the handbook was developed to provide initial training to the newly assigned configuration manager, it was decided that the paragraphs on allocated baseline (in Section 5 of the handbook) and System Design Reviews (in Section 6 of the handbook) should stay as written, following the policy stated in AFR 14-1. Thus, the newly assigned individual would be provided with what the official policy is, and if required, the individual program offices could relate why they have deviated from this policy based on their specific program's technical complexity, schedule constraints, and/or cost constraints.

The second recurring response from this second group was that they wanted the handbook (and would take it as is, without any further corrections/additions) as a Primer to use in training their "newer" configuration managers/specialists. They all concurred that the handbook would make for better understanding of the principles that comprise configuration management than is possible with the current approach of requiring their personnel to read the current military regulations, standards, and pamphlets that relate to configuration management. Each suggested that they would use the handbook to allow their personnel to become familiar with the CM principles, and then provide addendum to the handbook sections that would show how their program office specifically applies these principles. Each asked for, and will be provided with, a copy of the handbook to establish their own individual training programs.

The third group of individuals (the working level configuration managers/specialists) stated that they would have benefitted from having the handbook available to them when they first entered the program office. They agreed that trying to read the individual military documents, without having some other document that has tied these documents together, is not the most beneficial approach for the new configuration manager. It seems that more time is "wasted" trying to get other configuration managers to explain what the documents are "really" saying than is beneficially used obtaining a working knowledge of configuration management from the documents. So what is happening is that they are only reading the military documents in segments as they are assigned specific CM tasks. They agreed that having this handbook available to use as an introduction into the principles of configuration management, and as a follow-on useable reference, would still be of value to them.

V. Conclusions/Recommendations

Conclusions

Regardless of the sweeping changes that are taking place, and that will continue to take place in the future, in the acquisition environment, program managers will remain under continuous pressure to provide systems to meet the users' increasingly sophisticated performance requirements within time and cost constraints. Program managers can increase their probability of success by using configuration management as a technical management control system over those technical actions that occur during system design. After talking to configuration managers at Aeronautical Systems Division, it was determined that there existed a need for a document that describes the principles of configuration management that could be used to provide initial training to configuration managers.

Based on the comments received back from the configuration managers at the three different levels of experience, there now exists a handbook that can be used to provide initial training on, and an understanding of, the principles that comprise configuration management. This handbook (titled "Configuration Manager's Handbook: With Applications for Full-Scale Development," is included as Appendix HB) also provides guidance on how to apply the configuration management principles to a program that is entering, or proceeding through, the full-scale development phase of the system acquisition life cycle. This handbook will be reproduced and provided to the participating program offices for their use, and it will also be provided to ASD/ENC for distribution to any other program office that may want a copy.

Recommendations

To improve the contents of this handbook, there are several areas where additional research and evaluation could be performed.

1. The handbook could be provided to individuals being newly assigned to the configuration management section of different program offices. They would be requested to provide their inputs as to its readability and its clarity in providing insights as to the role of configuration management in the system acquisition process. These inputs could then be used to revise the handbook contents.

2. The handbook could be provided to individuals attending Professional Continuing Education classes (e.g., SYS 028 and SYS 228) provided by the Air Force Institute of Technology. Since individuals attending these courses come from other Air Force commands (and sometimes other services) their inputs would be beneficial as to the applicability of the handbook in other Air Force/Government areas besides Aeronautical Systems Division. These inputs could also be used to upgrade the contents of the handbook.

3. Additional research could be performed to determine what should be included in the handbook to assist configuration managers develop a CM program for a product entering any of the other system acquisition life cycle phases. These inputs could be used to develop separate sections (one each covering the concept demonstration/validation and production/deployment and operational support phases) which could be added to the handbook to provide guidelines for the configuration management practices to be used in that acquisition phase.

CONFIGURATION MANAGER'S HANDBOOK

WITH APPLICATIONS FOR FULL - SCALE DEVELOPMENT

PREFACE

This handbook is intended to assist Air Force program offices and configuration management personnel apply the principles of configuration management to a product under development. It briefly discusses the system acquisition life cycle as the domain in which a program is developed and the role of systems engineering in the development and design of the product. Configuration management is then introduced as the technical management control system that oversees the actions undertaken during the systems engineering process. The handbook then proceeds to introduce, and in subsequent sections describe, the four processes that comprise configuration management: configuration identification, configuration audits, change management, and configuration status accounting. Finally, the uses of these processes for the program in full-scale development are discussed.

TABLE OF CONTENTS

Page

Page

Page

Page

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

ACI	Allocated Configuration Identification
ACSN	Advanced Change Study Notice
AFLC	Air Force Logistics Command
AFR	Air Force Regulation
AFSC	Air Force Systems Command
AFSCP	Air Force Systems Command Pamphlet
ALC	Air Logistics Center
AQL	Acceptable Quality Level
ASD	Aeronautical Systems Division
ATP	Acceptance Testing Procedures
BSD	Ballistic Systems Division
CAGE	Contractor and Government Entity
CCB	Configuration (Change) Control Board
CCBD	Configuration Control Board Directive
CCP	Contract Change Proposal
CD	Classification of Defects
CDR	Critical Design Review
CDRL	Contract Data Requirement List
CGADS	Computer Generated Acquisition Document System
CI	Configuration Item
CIDS	Critical Item Development Specification
CII	Configuration Item Identification
CM	Configuration Management
CMP	Configuration Management Plan
CPCI	Computer Program Configuration Item
CPIN	Computer Program Identification Number
CRLCMP	Computer Resources Lifecycle Management Plan
CSA	Configuration Status Accounting
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
CSU	Computer Software Unit
DCP	Decision Coordinating Paper
DID	Data Item Description
DOD	Department of Defense
DOD-STD	Department of Defense Standard
DT&E	Development Test and Evaluation
ECP	Engineering Change Proposal
EDMO	Engineering Data Management Officer
ESD	Electronic Systems Division
FAR	Federal Acquisition Regulation
FCA	Functional Configuration Audit
FCI	Functional Configuration Identification
FQR	Formal Qualification Review
FSA	Functional System Audit
FSD	Full-Scale Development
HQ USAF	Headquarters United States Air Force
HWCI	Hardware Configuration Item
ICD	Interface Control Drawing/Document
ICWG	Interface Control Working Group

IDD	Interface Design Document
IOC	Initial Operating Capability
IRS	Interface Requirements Specification
MIL-STD	Military Standard
MNS	Mission Need Statement
MRB	Material Review Board
MSD	Munitions Systems Division
NOR	Notice of Revision
NSN	National Stock Number
NTE	Not-to-Exceed
OC-ALC	Oklahoma City Air Logistics Center
OI	Operating Instruction
OT&E	Operational Test and Evaluation
PCA	Physical Configuration Audit
PCI	Product Configuration Identification
PCO	Procuring Contracting Officer
PDR	Preliminary Design Review
PEO	Program Executive Officer
PIDS	Prime Item Development Specification
PM	Program Manager
PMD	Program Management Directive
PMP	Program Management Plan
PMRT	Program Management Responsibility Transfer
PRR	Production Readiness Review
RAS	Requirements Allocation Sheet
RFD	Request for Deviation
RFP	Request for Proposal
RFW	Request for Waiver
ROM	Rough-Order of Magnitude
SAE	Senior Acquisition Executive
SCCSB	Software Configuration (Change) Control Sub-board
SCN	Specification Change Notice
SCP	System Concept Paper
SDD	Software Design Document
SDP	Software Development Plan
SDR	System Design Review
SON	Statement of Operational Need
SORD	System Operational Requirement Document
SOW	Statement of Work
SPS	Software Product Specification
SRR	System Requirements Review
SRS	Software Requirements Specification
SSD	Space Systems Division
S/SDD	System/Segment Design Document
SSR	Software Specification Review
SSS	System/System Segment Specification
TCP	Task Control Proposal
TCTO	Time Compliance Technical Order
TDP	Technical Data Package
TDY	Temporary Duty
TRR	Test Readiness Review

1. INTRODUCTION

1.1 Purpose and Scope.

For most programs, the Government's program office establishes an internal organization (for major programs, usually called the Directorate of Configuration Management) to provide the structure for accomplishing the activities associated with configuration management (CM). This CM organization is staffed with configuration managers and configuration specialists (who will also be termed configuration managers for the purposes of this handbook) who should be the authorities on the CM practices and policies (as stated in the military regulations, standards, and pamphlets) for the program. They are responsible for setting up the internal CM practices for the program office and for establishing the CM requirements in the contract. However, the configuration managers in the program office and in the contractor's organization will require participation by many other functional specialists in order to effectively implement CM for the program. So everybody associated with a program (including those personnel in the Government's program office and those in the contractor's organization) should be aware of, and involved with, the use of the configuration management discipline during the program's system development.

The purpose of this handbook is to assist the configuration managers (and other program office personnel as required) in interpreting and applying the principles and requirements of CM to a product during the system acquisition life cycle. It is intended to be used primarily as a training instrument for those new configuration managers being assigned to programs that are about to (or already have) enter(ed) full-scale development. However, for those new configuration managers who are being assigned to a program in a different phase of the system acquisition life cycle, the first eight

sections of this handbook can still be used for general guidance in understanding the role of CM in the program office.

Configuration management can be thought of as a technical management control system. It is used by the program office as a means to monitor and implement the results of technical decisions made during system acquisition. The technical procedures that the CM discipline is used to monitor is referred to as systems engineering. For those who need to review the systems acquisition life cycle (including a review of the five phases a program may encounter during its acquisition), the function of a program office during system acquisition, and the role of systems engineering in acquiring the product, you may want to read Section 3, System Acquisition Process first. Section 3 is used to provide the framework and background for which CM was developed to work within.

For those who feel comfortable with the system acquisition process and the role of systems engineering, you may want to skip Section 3, and proceed straight to Section 4, Configuration Management - An Overview. Beginning with Section 4, and proceeding through Section 8, general guidelines on the CM discipline are presented, together with specific guidance in the basic CM areas of configuration identification, configuration audits (and design reviews), change (configuration and change control) management, and configuration status accounting. Section 9 is used to present suggestions for the configuration manager to apply the CM processes to a program that is about to enter (or has already entered) full-scale development. It presents guidance (including suggested Operating Instructions) as to what the configuration manager should be providing internally to the program office to ensure a successful configuration management program. In addition, Section 9 also provides suggestions as to what the configuration manager may want to include in the Request for Proposal

to ensure that the contractual CM requirements needed for the program are levied on the contract.

The Air Force's current CM policies and practices, as reflected by applicable Government regulations, specifications, and standards, have provided the major directions for this handbook. Some of these documents (those that should be accessible to the configuration manager in a program office) are presented, with a brief description of their contents, in Section 2, Relevant Documents. If the reader would like to obtain more detailed information on any of the subjects associated with a configuration management process discussed in this handbook, it is suggested that you review the descriptions of the contents of the documents related to that particular CM process as contained in Section 2 to determine which document would be the most likely to provide these desired additional information.

1.2 Contents of This Handbook.

1.2.1 Section 1, Introduction.

Describes the purpose and scope of the handbook, provides information as to which section the reader should begin with, and outlines the contents of each section.

1.2.2 Section 2, Relevant Documents.

Surveys the Government regulations, specifications, and standards relevant to CM on AFSC programs. In particular, this section addresses those Air Force Regulations (AFRs), Department of Defense/military standards (DOD-STDs and MIL-STDs), and AFSC Pamphlets (AFSCPs) that the configuration manager may find accessible in the program office to provide more detailed information than presented in each of this handbook's sections. These documents are listed according to their

applicability to program management aspects of CM, computer resources management, general CM, and the distinct areas of configuration management.

1.2.3 Section 3, System Acquisition Process.

Provides an overview of the program decisions, milestones, and phases of activity that a product must traverse to achieve the program objectives established by the appropriate authorities at the program's initiation. It also discusses the primary functions included in the program office that assist the program manager in overseeing the successful achievement of the program objectives. In addition, it outlines the systems engineering process in the procurement of a product.

1.2.4 Section 4, Configuration Management - An Overview.

Defines CM, discusses briefly the basic concepts of the CM processes, and describes the objectives and benefits of a successful CM program. The goals of CM associated with each phase of the acquisition life cycle are presented, and some of the possible tasks that may be used to achieve these objectives are outlined. Then some of the CM responsibilities of both Government and contractor personnel are discussed.

1.2.5 Section 5, Configuration Identification.

Discusses the aspects of the CM process of configuration identification. This section begins by discussing the concept of a configuration item (CI) in the acquisition of a system to include how CIs are selected; what are the consequences of selecting too many or too few CIs; providing a suggested checklist to use in deciding if an item should be considered a CI; and what are some document and management benefits of selecting an item as a CI. Then the concept of baseline management is discussed as it relates to configuration identification. The use of the three baselines (functional,

allocated, and product) within baseline management is discussed. In addition, the contractor's internal Developmental Configuration, as it relates to the control of a computer software CI, is discussed. Afterwards, the documentation that may be used to identify a CI is presented. The documentation discussed includes possible specifications and drawings that may be needed to document the design of a product during its development. Finally, the concept of CI numbering, those used for engineering/configuration control and those used for logistics/inventory control, is outlined.

1.2.6 Section 6, Design Reviews and Configuration Audits.

Briefly describes the functions of the systems engineering design reviews held during system development and the concerns of the configuration manager for these reviews. It then proceeds to outline the use of configuration audits to verify the system's design, to include a discussion of some of the items that need to be reviewed during these audits.

1.2.7 Section 7, Change Management.

Discusses the CM process of change management. This section begins by describing the configuration control aspect of change management. This includes descriptions of how engineering changes, deviations, and waivers are used to control the configuration identification and corresponding technical documentation for each CI designed/developed for the program. Next, the section discusses the change control (changes to contractual requirements not impacting technical baselines) aspect of change management. The section then examines the role of the Configuration (Change) Control Board (CCB) in the program office. It discusses preparations for the CCB, conduct of the CCB, and implementation of the CCB decisions. Finally, the

concept of interface control is introduced and briefly discussed. Although primarily a systems engineering function, configuration managers need to be aware of the documentation prepared and processed defining any interface requirements for the system/CI.

1.2.8 Section 8, Configuration Status Accounting.

Provides an overview of the configuration status accounting (CSA) function as a management information system. It then goes on to discuss the interrelations that exist between CSA and the other three CM processes. The CSA responsibilities of program participants are described, and finally, the depth of information required prior to, and after, the establishment of the product baseline is addressed.

1.2.9 Section 9, Applying CM for Full-Scale Development.

This section provides information, and suggestions, for the configuration manager to apply the four CM processes for a program as it enters (and/or progresses through) full-scale development of the system acquisition life cycle. Configuration management activities are required of the contractor and the program office's functional manager(s) to ensure successful technical management of the system/CI development. The CM activities that the configuration manager and other functional managers could perform for the program manager within the program office are discussed. Included are sample inputs for the Program Management Plan and for Operating Instructions to assist program office personnel perform configuration management activities. Then the configuration management requirements that should be tasked of the contractor are discussed. Included are suggestions for tailoring the requirements and sample Statement of Work tasking paragraphs and applicable Data Item Description numbers that can be used to request the deliverable data associated with these taskings.

2. RELEVANT DOCUMENTS

Configuration management (CM) is discussed in many Department of Defense (DOD) (including individual agencies) regulations, specifications, and standards. These documents range from those that discuss CM as one of a number of disciplines that are involved in the program management of the entire system acquisition to those that are devoted entirely to the various functions of configuration management. The following paragraphs discuss those military documents (in particular Air Force Regulations (AFRs), DOD military standards (DOD-STDs and MIL-STDs), and Air Force System Command (AFSC) Pamphlets) that can be used by Air Force program or configuration managers to supplement Sections 3 through 8 of this handbook if more detailed information is required in a particular area to better assist them apply CM to their product under development.

2.1 Program Management Documentation.

2.1.1 AFR 800-2, Acquisition Management: Acquisition Program Management, 16 September 1985 (With AFSC Supplement 1, 8 August 1986.

This regulation prescribes the system acquisition process for programs funded primarily through procurement appropriations or Research, Development, Test, and Evaluation appropriations. It applies from initial identification of mission need, through concept definition/exploration, concept demonstration/validation, full-scale development, and production/deployment. It provides basic policies and principles for the acquisition strategy of a product by the program office. It implements the applicable sections of DOD Directive 5000.1 and DOD Instruction 5000.2.

2.1.2 AFSC Pamphlet 800-3, A Guide for Program Management, 9 April 1976.

This pamphlet [which was under revision at the time of this writing] describes the general subjects and considerations that are involved in managing the acquisition of Air Force systems and associated elements. It does not specify a single, inflexible procedure through which all program goals are achieved. Instead, it is intended to assist (by providing a series of activities that are typically involved in the acquisition of systems) program managers, program office personnel, and others involved in the acquisition process to understand the process better and to help them plan and accomplish their assigned functions and responsibilities. The first five chapters describe the system acquisition life cycle and identify events and activities normally occurring during each phase. The remaining chapters discuss the major areas (including configuration management) involved in managing acquisition programs. The current version has few specific references to computer resources or software, but it is generally applicable to software development programs.

2.2 Computer Resources Management Documentation

2.2.1 AFR 800-14, Acquisition Management: Lifecycle Management of Computer Resources in Systems, 29 September 1986.

This regulation establishes the policy for the acquisition and support of computer resources (those employed as dedicated elements, subsystems, or components of systems) acquired under the program management concept of AFR 800-2; systems undergoing modification under AFR 57-4; or prototypes and demonstrations of advanced technologies that may be candidates for use in systems that will be acquired under AFR 800-2. It ensures that the computer resources involved in systems are planned, developed, acquired, employed, and supported to effectively and efficiently

accomplish assigned Air Force missions. It discusses those plans that should be requested of the contractor to develop and maintain the computer resources configuration during its development.

2.2.2 DOD-STD-2167A, Defense System Software Development, 29 February 1988.

This standard establishes uniform requirements to be applied during software development as it occurs throughout the system acquisition life cycle. It is not intended to specify or discourage the use of any particular software development practices, but rather it should be used to provide the means for establishing, evaluating, and maintaining quality in the software and the software's associated documentation. The requirements of this standard apply: (1) to the development of computer software configuration items (CSCIs), (2) to the development of the software element of any hardware configuration item (CI). (3) to any Governmental agency that performs software development, and (4) selectively to software not identified as a CSCI, but that the Government (or contractor) still requires to be developed in the manner described in this standard.

2.3 Configuration Management Documents

2.3.1 AFR 14-1, Configuration Management, 1 December 1988.

This regulation establishes the Air Force's policy for life-cycle configuration management of Air Force procured systems and configuration items, and for those joint programs that the Air Force is listed as the lead service. [NOTE: This includes subsystems, equipment, computer software, and other designated items.] It also prescribes uniform CM requirements, applications, objectives, and definitions for use throughout the product's life cycle.

2.3.2 <u>AFSC Pamphlet 800-7, Configuration Management, 1 December 1977.</u>

This pamphlet provides guidance on configuration management practices to all AFSC personnel who work with this functional area. The document contains philosophy, methods, and procedures based on DOD and Air Force policy. [NOTE: Although this pamphlet is still in effect, the material in it is dated. This document would be best used to get acquainted with the distinct areas of CM. Then, for each area addressed, the reader should read the other (more current) military documents that discuss the same area to insure that the policy has not been altered/changed since this document was published.] It describes concepts, techniques, and procedures for the efficient application of CM to systems, system segments, system modifications, equipment, computer programs, and other items designated as configuration items. This pamphlet also provides details of configuration identification, configuration control (which when combined with change control is referred to as change management), configuration status accounting, and information on design reviews, configuration audits, and computer software programs. It should be used with those military documents listed for each of the following configuration management processes (paragraphs 2.4 through 2.7).

2.3.3 <u>MIL-STD-483A, Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs, 4 June 1985.</u>

The purpose of this standard is to establish uniform configuration management practices that can be tailored to systems and configuration items procured by the Air Force and other agencies. It is a loosely structured collection of material, intended to be used with the documentation associated with each individual configuration management process, that establishes requirements in the areas of configuration identification, control, and audits; interface control; and configuration management

plans, reports, and records. Portions of this standard have been incorporated into MIL-STD-480B (described in paragraph 2.6.1) and at the time of this writing, material from MIL-STD-483 is being incorporated into MIL-STD-973 (which is currently in draft form and will be issued sometime in the future).

2.4 Configuration Identification

In addition to portions of the documents listed in the preceding paragraphs, the following documents describe various aspects of the configuration identification process.

2.4.1 DOD-D-1000B, Drawings, Engineering and Associated Lists, 28 October 1977 (With Amendment 4 Dated 18 August 1987).

This military specification has been superseded by MIL-T-31000 (see paragraph 2.4.7). However, for most programs that started prior to June 1990, this specification will still be in effect. It prescribes the requirements for engineering drawings and associated lists (tabulations of pertinent engineering information pertaining to an item depicted on the drawing(s)) acquired in one of three Levels. These Levels are: (1) conceptual and developmental design (Level 1), (2) production prototype and limited production (Level 2), and (3) production (Level 3).

2.4.2 DOD-STD-100C, Engineering Drawing Practices, 22 December 1978.

This military standard provides drawing practices for the preparation of engineering drawings and drawing format material. It includes: (1) a list of standards used for the preparation of engineering drawings and associated lists; (2) definitions and examples of types of engineering drawings to be prepared for the Department of Defense; (3) procedures for the creation of titles for engineering drawings; (4) numbering, coding, and identification procedures for engineering drawings, associated lists, and

documents referenced in them both; (5) methods for revising, and recording these revisions, engineering drawings; (6) requirements for the preparation of associated lists; and (7) criteria for changing part numbers (sections 402.14 and 402.15).

2.4.3 MIL-N-7513F, Nomenclature Assignment, Contractors Method for Obtaining, 14 November 1980.

This specification establishes requirements for the contractor in obtaining assignment of nomenclature for airborne, electronic, and aeronautical and support equipment. It should be used with MIL-STD-196, Joint Electronics Type Designation System; MIL-STD-875, Type Designation System for Aeronautical and Support Equipment; and MIL-STD-1661, Markings/Modification Designation.

2.4.4 MIL-S-83490, Specifications, Types and Forms, 30 October 1968.

This specification prescribes the general requirements for the degree of compliance with the requirements of MIL-STD-490 (see paragraph 2.4.6) in the preparation of program peculiar specifications. It briefly states the required contents and intended uses of the different types of general specifications. The specification also states the requirements and intended uses for the different "forms" of specifications. This information is currently being incorporated into a revision of MIL-STD-490, after which MIL-S-83490 will no longer be included on new programs.

2.4.5 MIL-STD-130E, Identification Marking of U.S. Military Property, 5 August 1977.

This standard establishes the item marking requirements and methods for identification of items of DOD military property. It provides both general and detailed marking requirements.

2.4.6 MIL-STD-490A, Specification Practices, 4 June 1985.

This standard establishes uniform practices for the preparation, interpretation, change, and revision of program-peculiar specifications. The standard is divided into a main body and 15 appendices. The main body states: (1) the broad requirements, concepts, and practices applicable to specifications in general; and (2) general requirements for each of the six sections of a specification. The appendices invoke the detailed requirements for the various types and subtypes of specifications.

2.4.7 MIL-T-31000, Technical Data Packages, General Specification for, 15 December 1989.

This specification now replaces DOD-D-1000B for new programs. It prescribes the requirements for preparing a technical data package (TDP) composed of one or more TDP elements and related TDP management data products. Appendix A of this specification provides guidance on determining what technical data should be acquired. Appendix B of the specification prescribes requirements for preparing and maintaining quality assurance provisions, if they have been specified in the contract, for the technical data package or any element of the TDP.

2.5 Design Reviews and Configuration Audits

In addition to portions of the documents listed in paragraphs 2.1 through 2.3, the following document describes various aspects of design reviews and of the configuration audit process.

2.5.1 MIL-STD-1521B, Technical Reviews and Audits for Systems, Equipments, and Computer Software, 4 June 1985 (With Notice 1 Dated 19 December 1985).

This standard prescribes the requirements for the conduct of technical reviews and configuration audits. It identifies the responsibilities of both contractor and Government

personnel. Each review and audit is generally described in the main body of the standard and more specifically defined in separate appendices. These appendices present outlines of the minimum required information to be available and verifications to be performed.

2.6 Change Management

In addition to portions of the documents listed in paragraphs 2.1 through 2.3, the following documents describe various aspects of the change management process.

2.6.1 MIL-STD-480B, Configuration Control - Engineering Changes, Deviations, and Waivers, 15 July 1988.

This standard establishes the requirements, formats, and procedures to be used in the preparation of configuration control documentation. It provides requirements for: (1) maintaining configuration control of both hardware and computer software configuration items; and (2) preparing and submitting engineering change proposals (ECPs), deviations, waivers, notices of revision, and specification change notices. It should be imposed on contractors who have participated or are participating in the engineering or operational development of a system or high-level configuration item. In addition, it may be imposed on those contractors who are being supplied with copies of the system specification and development specification(s). The requirements established by this standard apply only to the functional, allocated, or product configuration identifications approved (baselined) by the procuring activity.

2.6.2 MIL-STD-481B, Configuration Control - Engineering Changes (Short Form), Deviations, and Waivers, 15 July 1988.

This standard establishes requirements, formats, and procedures for the preparation, submittal, and approval or disapproval of engineering change proposals,

deviations, and waivers for those contracts that involve multi-application items not peculiar to a particular system. It is used for contractors who cannot reasonably be expected to know the complete consequences of the engineering change on all units and support elements affected by the change. The procuring activity performs much of the impact analysis when this standard is applied.

2.7 Configuration Status Accounting

In addition to portions of the documents listed in paragraphs 2.1 through 2.3, the following document describes various aspects of the configuration status accounting process.

2.7.1 MIL-STD-482A, Configuration Status Accounting Data Elements and Related Features, 1 April 1974.

This standard provides a comprehensive listing of standard data elements and their related data codes, use identifiers, and data chains that should be used in tailoring the content of configuration status accounting records and reports. It does not prescribe which data elements and related features to use, nor the format or frequency of submittal of these reports. These decisions are decided upon by the procuring activity and contractor during negotiations.

3. SYSTEM ACQUISITION PROCESS

This handbook is intended to assist the configuration manager in performing the configuration functions for a single product program office. In most cases, for a product to be assigned to its own program office it will be, or has been, designated as either a major system acquisition program or a Component acquisition program. The difference between these two is the level of milestone decision making. A program that has been designated as a major system acquisition program requires a Secretary of Defense decision at each of its milestone reviews, while a Component program has had the authority of its milestone decisions delegated to the appropriate DOD Component Head. This section will provide the configuration manager a brief review of the system acquisition life cycle, the functions associated with the program office, and a review of the systems engineering process.

3.1 System Acquisition Life Cycle.

The system acquisition life cycle is a sequence of key program decisions, milestones, and phases of activity directed towards the achievement of program objectives which were established by the approving authorities at the program's initiation and recorded in the Program Management Directive (PMD). A program's transition through the system acquisition life cycle, begins with the generation of the Statement of Operational Need and results in the operational deployment and support of the weapon system that can meet that stated operational need. This process includes both the acquisition of the specific mission equipment, and the acquisition of such peripheral elements as the common and peculiar support equipment, technical data and manuals, spares, facilities, computer resources, etc.

This system acquisition life cycle is normally divided into five phases to enhance management effectiveness (Figure 1). Since each product has its own unique features, no two system acquisition programs are identical. Thus, each of the acquisition phases should be tailored to minimize acquisition time and life-cycle costs, consistent with the urgency of need and degree of technical risk involved. Each phase

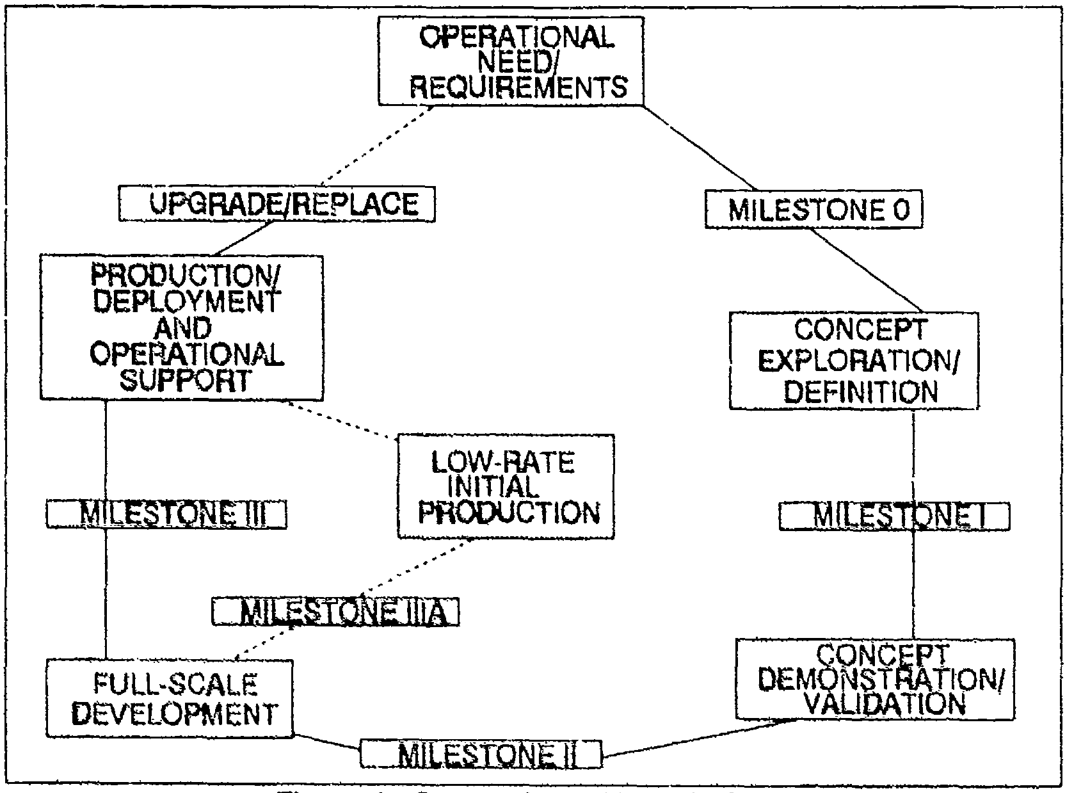

Figure 1: System Acquisition Life Cycle

may involve an iterative process to ensure that the output from that phase is optimized. However, depending on the magnitude and cost of the acquisition, some of the phases may be combined or omitted. The point in the life cycle at which a program enters, after receiving its initial approval, is also dependent upon the maturity of the technology involved and the uniqueness of the concept. The outputs from each phase should provide a definitive and documented baseline for the entry of the program to its

next phase. The underlying objective of each phase is to develop a level of confidence in the solution being offered to meet the operational need and to reduce the risks involved in proceeding to the next phase.

At the end of each phase, the results obtained and the risk involved in continuing the program are reviewed and compared to the initial objectives that were derived from the operational requirement. The results of this review is recorded by updating the PMD. Direction will be provided to either continue, modify, transfer, or terminate the program. The discussion that follows concerning the phases involved in the system acquisition life cycle is oriented towards the procurement of major weapon systems. However, smaller programs must proceed through essentially the same process with more combined/omitted phases and with a lower level of review/decision authority.

3.1.1 Operational Need/Requirements Phase.

The overall purpose of the system acquisition life cycle is to provide a weapon system that satisfies an operational requirement. The process starts with the identification of an operational need or operational deficiency (resulting from threat changes, redefinition of assigned tasks in response to a shift in national security policy, or deterioration of older systems) that cannot be satisfied through changes in tactics, strategies, doctrine, or training. This obligation has been delegated to the major operating/using commands where mission area responsibilities rest, and is outlined in AFR 57-1, Operational Needs, Requirements, and Concepts. To perform this function, the major operating/using commands conduct continuing mission area analyses to identify mission element deficiencies in both present and projected capabilities and to identify present and projected threats.

If the operational need cannot be satisfied within the operating/using command's existing capabilities, or through changes in its tactics, strategies, doctrine, or training; the command can process a Statement of Operational Need (SON). This document describes each need in operational terms relating to both planned operations and support concepts and is also used to address potential solutions to the operational needs (to include both upgrades of existing systems and development of new systems). The operating/using command uses the SON to define the operational need; to document official validation of the need; and to furnish the implementing, supporting, operational test and evaluation, and other participating commands the preliminary requirements for these commands to use for their planning purposes. The SON is then coordinated among the operating and support commands for comments and the final version is submitted to HQ USAF, Deputy Chief of Staff/Plans and Operations.

At HQ USAF, the SON is reviewed and the stated need (requested program) is identified (with assistance from the Secretary of Defense's office) as either a major program, a Component [e.g., AF-managed] program, or an other-than-major program. This identification of the type of program is based on: (a) development risk, urgency of need, or other Secretary of Defense interests; (b) if the program is to be a joint acquisition of a system by two or more Department of Defense (DOD) Components or by the DOD and representatives of another nation; (c) the estimated requirement for research, development, test and evaluation (usually a threshold of $200 million in 1980 dollars) and/or procurement funds (usually a threshold of $1.0 billion in 1980 dollars); (d) estimated requirements for manpower to operate, maintain, and support the system in the field; or (e) Congressional interest in the stated program. For those

that are designated as either major or Component programs, the SON is used to prepare a Mission Need Statement (MNS).

Approval of the MNS (for major or Component programs) or the SON (for other-than-major programs) constitutes the program initiation decision point. This approval is referred to as a Milestone 0 - Program Initiation/Mission-Need Decision. Normally, for a program that might include advances in technology or an innovative management approach, a concept exploration/definition phase will follow the Milestone 0 decision. For those identified as other-than-major programs (normally those requiring development of systems using current technology, or modifications of existing systems), following a successful Milestone 0 decision, they will usually proceed straight to full scale development. Here the rationale is that there is no technological concept that needs to be explored or demonstrated; the development of the system will be confined to reasonably well-understood technical constraints and parameters.

3.1.2 <u>Concept Exploration/Definition Phase.</u>

With a favorable Milestone 0 decision, HQ USAF issues a Program Management Directive (PMD) to the implementing command (usually Air Force Systems Command, AFSC), and to all participating commands, which formally implements the decision. The initial PMD has two basic purposes. First, it notifies the originating operating/using command that it must develop a System Operational Requirements Document (SORD) for the candidate system. [NOTE: In those instances where the solutions are dissimilar and cannot be addressed in a single document, more than one SORD may be required.] The SORD is used by the respective operating/using command to describe pertinent quantitative and qualitative performance, operation, and support parameters, characteristics, and requirements for a specific candidate weapon system.

The SORD will also provide guidance to the implementing, supporting, and other participating commands by documenting how a system will be operated, deployed, employed, and supported. At each subsequent milestone decision, the responsible command will update and refine the SORD(s) to reflect changes in the need, evolving requirements, or cost and performance tradeoffs.

The second purpose of this initial PMD is to direct the start of the concept exploration and definition phase effort. It provides stated user/HQ USAF requirements and requests various studies to be performed. One of the most important elements of the PMD that is issued for this phase is the assignment of a program manager (PM), the issuance of a charter stating the PM's responsibility and authority, and the establishment of a program office.

Entering this phase of the system acquisition life cycle does not automatically commit the service to the development and acquisition of a new system. The concept exploration/definition phase represents a commitment to go only so far as to identify and explore possible alternative solutions to the stated operational need. These alternative solutions may include new system acquisition efforts, remedies through doctrinal changes, modifications to existing equipment, resource allocation changes, or off-the-shelf procurement of a commercial system. To insure that all practical alternatives are considered, industrial contractors and federal laboratories/research centers are solicited for system design concepts. The document that is used to solicit proposed solutions is the Request For Proposal (RFP). The RFP should provide complete information on the mission need, the operating environment and threat, schedule and cost goals, and capability objectives. The offerors then propose their approach and main design features back to the Government for review.

After all the proposals are received by the program manager, the proposed alternatives are evaluated by the program office personnel for technical, support,

operational and maintenance concepts, and life cycle cost. Based on this evaluation, a contract is awarded to one or more contractors/contractor teams to continue their exploration/definition efforts. The type of analyses that may be included during this phase can range from the pure paper study to building prototypes for limited experiments and tests. When this phase is completed, the program office prepares documentation to support the program review and corresponding Milestone I decision that is required prior to the program entering the next system acquisition life cycle phase. The primary document used to present the findings during the concept exploration/definition phase is the System Concept Paper (SCP). The format and content of the SCP is provided in DOD Instruction 5000.2, Defense Acquisition Program Procedures, 1 September 1987, Enclosure 4. This document provides the decision making authorities a brief review of the relevant mission area and role of the operational requirement, the threat assessment, and the shortfalls of existing systems within the mission area to meet the threat. The SCP also addresses the capabilities of the alternatives considered by the program office, a description of the selected alternative(s), and a technological risk evaluation for each of the selected alternatives. At this milestone decision point, Milestone I, the program is either approved to enter the next phase or canceled; or in some cases, requested to repeat some of the effort from this phase and return back for a decision after the additional tasks have been performed. The primary factors that influence the decision are the achievability of the alternative(s), affordability, reasonable schedule, appropriateness of the proposed solution, and continuing need for the capability.

3.1.3 Concept Demonstration/Validation Phase.

With a Milestone I approval, and the issuance of the corresponding revised PMD by HQ USAF, the system acquisition life cycle enters the concept demonstration/

validation phase. The primary thrust of the effort during this phase is the reduction of technical risk and economic uncertainty by focusing on refining the selected alternative(s) through studies and analyses and may include hardware development, limited tests, and evaluations. Typically, the alternatives are evaluated in one of three design approaches: (1) a system design definition study, (2) system prototyping, or (3) a combination of system definition and prototyping.

3.1.3.1 Design Definition. This validation approach utilizes competition between alternatives using system studies and analyses to define the proposed system designs. These designs are expanded in their definition of the system to provide a more concise technical, cost, and risk data about the specific design approach that would be used to achieve the stated requirement by delineating the system requirements down to its configuration items. These configuration items are those items of the proposed design, designated by the program office, whose performance parameters and physical characteristics must be separately defined, specified, and controlled to provide the program manager the insight needed to achieve the overall end use function and performance of the proposed design.

This specific design approach is continuously reviewed for its operational value and its practicality. The alternative design approaches are then evaluated by the program office and the using command, or a source selection review team, based on their risks, tradeoffs required, and resources required. The products of this approach are system and initial hardware configuration specifications, refined cost estimates, and schedule projections.

3.1.3.2 System Prototyping. This is the most frequently used, and currently preferred, approach for demonstrating and validating the candidate system concepts. This approach entails the maintaining of at least two contractors to build prototypes of their

system, which can then be evaluated towards the end of this phase of the acquisition life cycle. If only one of the alternative concepts was recommended for this phase, then that concept can be contracted out to at least two contractors who will build their "system" to meet this concept. If on the other hand, more than one alternative was approved to enter the concept demonstration/validation phase, then the prototypes that embody these different alternative solutions will be produced. With the prototypes developed, they are then competed against each other based upon some competitive performance evaluation criteria. The competition is evaluated by the program office based upon the system being able to meet the performance objectives but not necessarily other operational system characteristics. Again, the evaluation process that follows will not only include the results of the above competition, but will also review (among other requirements) the corresponding risk associated with each alternative, its operational suitability and effectiveness, the resources required, and life cycle cost.

Upon completion of either of these two individual approaches, or some combination of them, the program office prepares to once again enter the acquisition review and approval process. The primary document that is used to provide the results of this phase is the Decision Coordinating Paper (DCP). The format and content for the DCP is provided in DOD Instruction 5000.2, Defense Acquisition Program Procedures, 1 September 1987, Enclosure 4. This document readdresses the mission area and role, the corresponding threat assessment and the shortfalls of existing operational systems to meet this threat, the continuing need for this requirement, the alternatives considered, and a description of the selected alternative(s) carried into the demonstration/validation phase. The DCP provides the results of the evaluation process performed on each of the alternatives during the demonstration/validation phase. The decision authorities then review these results to determine if there is

sufficient evidence provided to support a decision that a satisfactory solution has been demonstrated and validated. Note that the review authorities are not presuming that there are no risks or uncertainties remaining in the program. On the contrary, they are reviewing the program to ensure that any remaining technical, cost, and schedule risks have been identified and can be resolved within program resource and time constraints. Given that they agree with the program office's assessment, they will then approve the program to move into the next phase. If these decision authorities do not agree that the program risks are solvable within achievable program resources and time constraints, or if the need no longer exists, the program has a high probability of being canceled.

3.1.4 Full-Scale Development/Low Rate Initial Production Phase.

An approval at the Milestone II decision reaffirms the operational need, approves the selection of one (or more) of the competing designs to continue through engineering development and may (for some programs) authorize ordering of long-lead procurement items and allow the initiation of low rate initial production to verify production capability and to support limited operational test and evaluation. During this phase, the system, and all principal support items and documentation, is designed, developed, fabricated, tested, and evaluated. The intended output is a preproduction system (unless low rate initial production was begun, then the output could be a production system) that closely approximates the final operational product, the documentation needed to enter the production phase, and test results that demonstrate that the design meets the stated requirements.

By the end of this phase, detailed design development specifications should be finalized and engineering drawings completed. These specifications and drawings will become the performance basis for production/acceptance of the quantities of units

delivered during the production phase. [NOTE: For most major programs, only preliminary product specifications and/or production drawings will exist during this phase.] One of the major efforts of this phase is the fabrication of the pre-production prototype hardware and/or software test items. The Government monitors this effort, including the generation of the specifications and design documents, throughout system development with technical reviews that grow increasingly more detailed as the program design evolves. In addition, the program office may also become involved in the review of mockups of the proposed system to assure that the mockup configuration being built appears capable of meeting the current approved design requirements. This review process culminates at the Critical Design Review(s), where the Government has its last chance, without adding significant cost and time to the development, to communicate their concerns about elements of the system design (which may not meet the stated operational requirements) before the design is committed to the manufacture of hardware or the coding of software computer programs.

In addition to the design reviews, system and component testing is a vital ingredient of a successful full-scale development program. System testing begins with the early development of plans for the testing of various components that have been identified as program risks, but it is primarily focused on the testing of the configuration items that comprise the system to verify that they fulfill the specified requirements. Throughout the testing process, design problems are identified, assessed, and reduced; operational effectiveness/suitability is evaluated; and design deficiencies are identified and corrected. The test results become more significant as each configuration item proceeds through development testing to the final full system operational tests with all appropriate support elements of the system. The two types of

testing encountered to meet this requirement are development test and evaluation (DT&E) and operational test and evaluation (OT&E).

3.1.4.1 DT&E. This is essentially the final detailed engineering verification of the system performance. DT&E begins with testing of the configuration items. Each of the configuration items is tested against the requirements in its specification, which specifies its portion of the overall system requirement. The DT&E process continues until all of the configuration items are joined together and the full system design is tested and evaluated, by the implementing command, against system requirements. The primary purposes of DT&E are to demonstrate that the design and development are complete (or to identify the deficiencies of the design to the program office), to show that the design risk has been minimized, and to demonstrate that the system, as currently designed, meets the specification requirements.

3.1.4.2 OT&E. This type of test is an operational assessment of the system during which it is evaluated against the stated operational criteria. The testing is performed by personnel who have the same speciality codes and skills as those individuals who will eventually operate, maintain, and support the system when it is deployed. The primary purposes of OT&E are to demonstrate that the system can be supported logistically in a deployment status and to assess the system's military utility, operational effectiveness, and suitability.

Upon completion of the testing and other full-scale development phase actions, the approval process is once again encountered. At this Milestone III point, the program is reviewed to reaffirm that the operational need still exists, that the proposed system is still the best alternative, that test results support the readiness of the system to enter production, that the production process is ready to support building the system, and that the system can still be acquired to a reasonable schedule and operated at a

reasonable cost. With a Milestone III approval, the program is now ready to enter the last phase of the system acquisition life cycle the production/deployment and operational support phase. A revised PMD is issued that provides the implementing and supporting commands with the direction to proceed into the next phase.

Some programs, faced with program schedule pressures to meet established initial operational capability dates, may decide to begin a low rate initial production program before the completion of testing of the design. This overlapping between developmental testing and initial production is referred to as concurrency and will normally require a Milestone IIIA approval. At this approval point, the acceptance of certain risks must be justified since the system testing (and sometimes the CI testing) will not be fully completed (or, in some instances, may not have even started) before this low-rate production process will begin. The major risk is whether the design is deemed, by review of the test results to that point in the program, to be so close to the final design that the production process can be established with little risk of major changes to be required to the design, to the production line, and to already delivered units. These decisions must be made judiciously in order to permit the concurrency when the expected benefits (e.g., program continuation or early system deployment) of starting the production process outweigh the potential costs (schedule and financial) if a test failure leads to a design revision which requires a change to the production process and to delivered units.

3.1.5 Production/Deployment and Operational Support Phase.

This phase is divided into three overlapping sub-phases: production, deployment, and operational support.

3.1.5.1 Production Sub-Phase. This is the time between the Milestone III decision and the delivery and acceptance of the last system. It includes the production of all system

hardware, spares, support equipment, data, software, etc. Early in this portion of the life-cycle, verification of specification compliance is completed and the operational tests that were begun in the last phase are usually concluded. Also during this sub-phase contractor participation in the system testing process is finished. However, even after the contractor's participation is completed, the system elements continue to be tested and integrated by Government personnel (operational test and evaluation personnel) until the system is as near to the designed/required operational configuration possible so that the system will reach operational maturity. That is, the system is tested to identify changes that will make the system operationally suitable and maintain currency to its mission need. Due to limitations in testing, the correct operational configuration may not be achievable in a test environment. If this is the case, the idea is to test until the appropriate test limitation is met.

Another important aspect of this sub-phase is the program management responsibility transfer (PMRT). This is the formal act of transferring program management responsibility, including engineering responsibilities, from the implementing command to the supporting command (and its responsible Air Logistic Center). PMRT should occur at the earliest practical date (that point when most developmental engineering and testing actions are successfully completed and action items from these events are closed) during the production sub-phase; it will usually occur even though a significant number of production units remain to be ordered and/or delivered.

3.1.5.2 Deployment Sub-Phase. This phase begins when the first production items are delivered to, and begin being used by, operational personnel. This will normally occur while the system is still in production, thereby resulting in a concurrent production/deployment phase. The primary focus of the deployment sub-phase is the

activation of the system in the field. During this phase, the system is turned-over to the using command. With the turnover, the using command formally accepts responsibility for the operation and maintenance of the delivered operational units of the new system and assumes property accountability. This point in the program's life cycle is referred to as its initial operating capability (IOC). The testing that has been associated with OT&E up to this point has been focused on evaluating the system against those current operational criteria associated with system operation, maintenance, and support. The using command will usually continue some form of OT&E testing after system turnover has been accomplished. However, the OT&E testing now involves identifying and investigating new operational uses for the system and developing/reshaping the most effective operational tactics and techniques that can be performed with the system.

3.1.5.3 Operational Support Sub-Phase. Deploying the system also marks the beginning of the operational support sub-phase. This sub-phase continues until the system is removed/retired from the active inventory. A formal review will normally occur between one and two years after the system is initially deployed to identify actions and resources required to insure the readiness and support objectives of the system are achieved and maintained. If the system is failing to achieve the contractually required standards (e.g., not able to totally perform the mission for which it was acquired), the Air Force needs to determine the deficiencies. If there are any warranties remaining in the production contracts, they should be exercised to correct these deficiencies. A second review will normally occur 5 to 10 years after initial deployment of the system to establish the need for major modifications or system replacement. This review determines if, with the system operating as required, it has the capability to meet mission requirements. If the system is falling short of current

needs, then investigations will be undertaken to determine if major modifications can be provided to enhance the system to meet the required mission or if the deficiencies are such that a new program should be undertaken to develop a replacement system This portion of the operational support phase is used to evaluate the changing mission analysis and successful mission accomplishment. New system development will begin when either technology or an operational need dictates that the process has completed a full circle.

3.1.6 Conclusion.

The preceding paragraphs briefly described the system acquisition life cycle. However, because every system developed has unique characteristics, the above system acquisition life cycle must be flexible. There is no set path that must be followed from a Statement of Operational Need to system deployment and operation. Any program, depending upon the technology involved, could begin at any of the milestones in the acquisition process and, if approved, might not have to proceed from the beginning of the related phase. In addition, because of uncertainties, unforeseen developments, or mission changes, a program could be required to remain in a phase or even redo some of the work from a preceding phase. Finally, since every program will face some type of review process at various points during its development, it is always in jeopardy of cancellation. Thus, just because the program enters the system acquisition life cycle does not insure that a new operational item will be forthcoming.

3.2 The Program Office.

3.2.1 Purpose and Initiation.

The program office is the focal point for involvement by all the agencies that participate in the acquisition of a new or modified system/product. In addition, it is the

only organization that is authorized to direct any of the contractor(s)'s efforts. Its primary mission is to get a system to the required user(s) that meets cost, schedule, logistic supportability, and performance requirements.

As mentioned in paragraph 3.1.2, the program office is initially formed during the concept exploration/definition phase of the system acquisition life cycle. With the formal "go-ahead" given with a Milestone 0 approval, HQ USAF provides formal direction to the implementing and participating commands with the issuance of a Program Management Directive (PMD). [For the Air Force, the implementing command is normally Air Force Systems Command (AFSC).] Upon receiving direction from HQ USAF via the PMD, HQ AFSC issues their formal direction, in the form of AFSC Form 56, to the appropriate product division.

[Note: At the time that this document was written, there was a Defense Management Review ongoing that was working to streamline the acquisition process. One of the measures being undertaken was the reduction in the number of management review layers between the program manager and the decision authorities. Under this realignment, the program manager is to report to, and receive direction from, a Program Executive Officer (PEO) who is responsible for administering a number of acquisition programs. The PEO in turn reports to, and receives direction from, a Senior Acquisition Executive (SAE) within the appropriate Military Department (designated by the Component Head) who is responsible for administering the acquisition programs in accordance with established Department of Defense policies and guidelines. However, even with this reduction in the number of management review layers, each remaining review authority will inevitably provide their own "official" direction to the program office. With this in mind, it is not known to what extent the AFSC Form 56s will be used in the future. In addition, at the time of writing of this document the AFSC Product Divisions include: Aeronautical Systems Division (ASD),

Electronic Systems Division (ESD), Ballistic Systems Division (BSD), Space Systems Division (SSD), and the Munitions Systems Division (MSD)].

The product division receiving the direction then determines whether the program will be managed as one of many assigned to an existing program office, or if a new program office will be established to manage just this product. This handbook is intended to address requirements/practices for the product that is being designed, developed, and supported within a program office responsible for that product only. Once it is created, the program office continues in existence until the program is terminated or until it is transferred (PMRT) to the supporting command. However, whether the program is canceled or transferred, residual tasks usually remain that will require the program office to continue as a management entity.

3.2.2 <u>Organization.</u>

To perform its mission, the program office consists of individuals from many different functional areas. In line with the philosophy that each product is unique in its own way, there is no such thing as a "standard" program office. Within the constraints specified by the PMD and AFSC Form 56, the program manager must tailor the internal organization of the program office to meet the objectives of the specific program/product. Some of the factors which affect this tailoring process are: (1) the acquisition strategy, (2) the program management concept, (3) the nature of the program (e.g., size, importance, cost and duration), (4) current phase of the system acquisition process, and (5) any manpower or other resource considerations.

For the single product program office, a common organizational structure is as shown in Figure 2. The figure shows a program office comprised of individual functional elements (often called directorates) which all work together to support the program manager and achieve the program office's overall program management

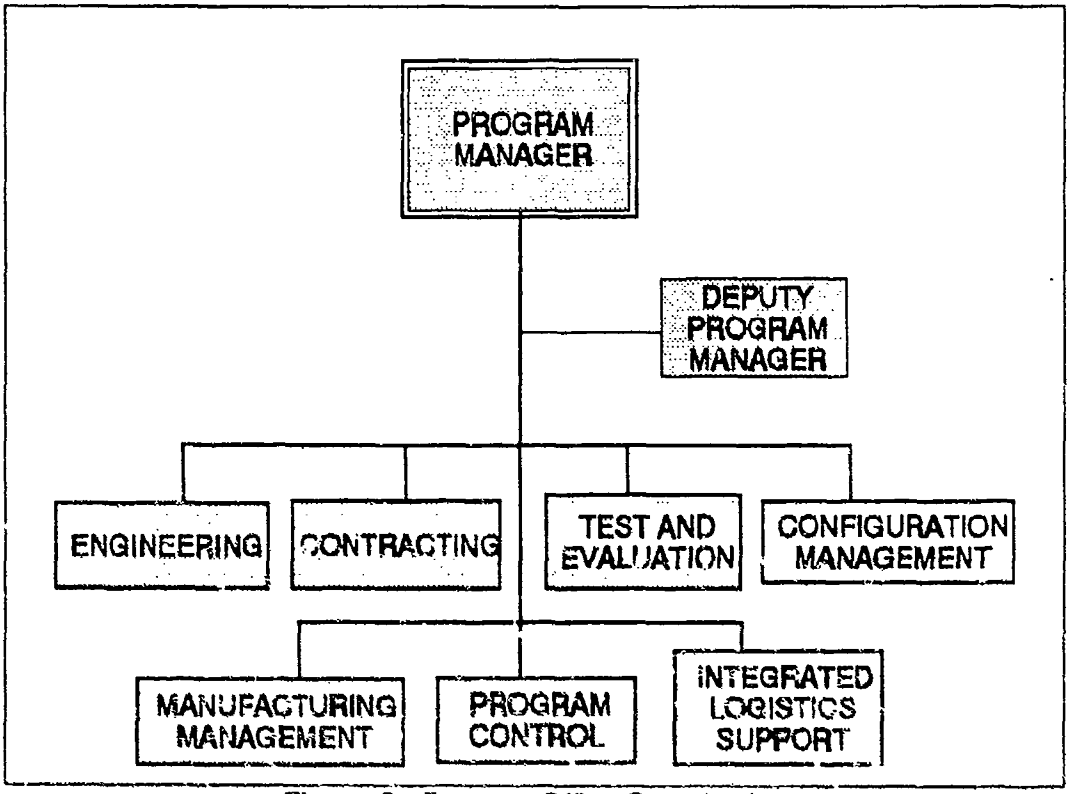

Figure 2: Program Office Organization

objectives. However, this structure is not necessarily the only way the functions associated with these directorates can be aligned. In some program office organizations, these functions can be grouped together in other arrangements to form directorates of some other name, which combine many of the functions that these individual directorates perform. The important point to realize is that these functions are aligned to meet program management objectives. How they are arranged is a program particular decision. Even so, in general, we can discuss how these functional elements support the program office mission.

3.2.2.1 The Program Manager. As seen in Figure 2, all the directorates are responsible to, and support, the program manager (PM). The PM is the individual who is responsible for the ultimate success or failure of the program. The PMD includes a

program charter which provides the PM the responsibility, authority, and accountability for achieving the stated program objectives. From the point of program initiation, until program management responsibility transfer (PMRT), the implementing command's program office program manager is the "manager-in-charge."

With the exception of Operational Test and Evaluation decisions, the PM is responsible for all technical and business decisions for the program, and these decisions are directive on all the program's participating commands. The PM is the focal point for communication from the contractors, Air Force field organizations, other major commands, and any other civilian or military organization concerned about/with the program. To assist with information flow to/from the contractor, the program organization must include a procuring contracting officer (PCO) who has official statutory authority to direct contractual actions to be performed by the contractor. In other words, the program manager is responsible for the total system program, provides direction to the contractor(s) through the PCO, and holds the program office personnel responsible for verifying the successful completion of specific tasks and objectives by the contractor.

3.2.2.2 Engineering Directorate. This directorate provides technical advice to the PM and manages/oversees the contractor's system engineering function (e.g., system and subsystem integration, reliability, maintainability, etc). It also provides system program technical direction (through the PM and/or the PCO) to the contractor, assesses the technical compatibility of all system elements, and verifies that the contractor designs, produces, and delivers a system that meets the stated requirements.

3.2.2.3 Integrated Logistics Support Directorate. This directorate, normally manned by both AFSC and AFLC personnel, provides the program manager with logistical/ technical guidance and assistance in logistic support elements. Some of these logistics support elements are in the areas of support equipment, maintenance, supply, manpower, and computer resource support. These and all the other logistic support elements are discussed in the PMD, in the Integrated Logistics Support section of the Program Management Plan, and in the Integrated Logistics Support Plan (which is used to plan, implement, and integrate the elements).

3.2.2.4 Configuration Management Directorate. This directorate is responsible for formalizing the system requirements into system and lower level design specifications, controlling both hardware and software configurations, and accounting for all configuration items. In addition, this directorate is normally responsible for data management and managing contract change management functions. It is through this group that the program manager is provided insight into, and maintains control over, the technical results of the program. Exact CM responsibilities will be explained later in this handbook.

3.2.2.5 Program Control Directorate. This directorate is responsible for program planning, progress tracking, trend analysis, and financial accountability. It is often referred to as "the nerve center" of the program office. Through the cost and schedule tracking provided by this directorate, the program manager is able to maintain total management control, surveillance, and overall program understanding.

3.2.2.6 Test and Evaluation Directorate. This directorate is responsible for planning, coordinating, and managing the overall system test effort (except for Operational Test and Evaluation as mentioned in paragraphs 3.1.4.2, 3.1.5.2, and 3.2.2.1).

3.2.2.7 Contracting Directorate. This directorate contractually implements, and manages the contractual accomplishment of, all the contracted acquisition activities required by the program office. The contracting officer, or specified representative, is the only individual(s) authorized to request performance of work by the contractor. Any request for additional work, clarification of performance required, or resolution of disputes must be coordinated through this directorate to avoid unauthorized contractual redirection that may result from these interactions with the contractor.

3.2.2.8 Manufacturing Management Directorate. This directorate is responsible to the program manager for all manufacturing activities involved in the acquisition process. These individuals assure that the product being developed to meet the stated program objectives is being designed for producibility. In addition, through the use of Production Readiness Reviews, the manufacturing management personnel are able to assess whether the procedures, techniques, and actions undertaken by the contractor in the production facilities will permit the contractor to produce an acceptable final product (e.g., one that meets the stated technical requirements) in the quantities/ schedule required by the program office.

3.3 The Role of Systems Engineering.

As discussed in paragraph 3.1, the system acquisition life cycle is the process through which the military services procure their weapon systems. This life cycle begins with the user's needs (both the constraints and the requirements needed to

satisfy operational mission objectives) and concludes when the system is retired from the inventory. To assure that the design decisions being made as the system is under development address the impact on all elements of the system and not just the immediate component being designed, the system design evolves through a process that is known as systems engineering. Configuration management complements this process by making sure that the systems engineering results are incorporated into appropriate design documentation and that the documentation is then placed under government control at an appropriate point in the life cycle.

3.3.1 Systems Engineering Overview.

Systems engineering is both a technical process and a management process that should begin from Day 1 of the system acquisition life cycle. Although developing systems are not identical, and each program differs in its underlying requirements, systems engineering provides a uniform (consistent), identifiable, and logical process to define the optimum system design and to efficiently accomplish the tasks associated with completing the design. During the early development (i.e., conceptual) period of a program, systems engineering is used in conceiving the design concept and in defining the requirements for the system. As the design development progresses, systems engineering assures that: (1) at each step of the design process, all system elements have been identified and that their related requirements and design are being addressed; and (2) as a part of an integrated design effort, impacts on all system elements are being considered. When development is completed, system engineering is concerned with verifying that the specified performance for the system and its sub-elements has been achieved. When the acquisition life cycle reaches the Production sub-phase, systems engineering is concerned with verifying that the system operating

capability has been achieved, evaluating any proposed changes to the system to determine the technical effectiveness of these changes, and to facilitate the incorporation of any changes, modifications, or updates.

3.3.2 Formal Definition.

The current application of systems engineering concepts, the requirements for military development programs, and the tasks involved in defining the systems engineering process are included in MIL-STD-499A, Engineering Management. This document defines systems engineering management as:

> ...the management of the engineering and technical effort required to transform a military requirement into an operational system. It includes the system engineering required to define the system performance parameters and preferred system configuration to satisfy the requirement, the planning and control of technical program tasks [that is, the management of those design, development, test, and evaluation tasks required to progress from an operational need to the deployment and operation of the system by the user], integration of the engineering specialties [that is, the timely and appropriate intermeshing of engineering efforts and disciplines such as reliability, maintainability, logistics engineering, human factors, safety, etc., to insure their influence on system design], and the management of a totally integrated effort of design engineering, specialty engineering, test engineering, logistics engineering, and production engineering to meet cost, technical performance and schedule objectives.

The process outlined in MIL-STD-499A contains general engineering management criteria that should be tailored to fit the particular needs of the system under development.

3.3.3 Systems Engineering Objectives.

The systems engineering process starts with the needs and/or requirements of system performance capabilities, and it then allows for the analyses of various design alternatives to meet the required performance capabilities. By the end of the analysis stage, the systems engineering process has attempted to optimize the design of the

system so that the resultant design addresses all of the system requirements. Finally, this process requires that the design decisions made are recorded in the appropriate design documentation. To facilitate the accomplishment of these objectives, the individuals involved in the systems engineering process:

(1) Identify the system requirements and translate them into basic functional requirements.

(2) Identify alternative functions and their associated performance and design requirements.

(3) Perform system and design engineering analyses of the alternative functions, with their associated design, personnel, training, and procedural requirements.

(4) Determine the optimum design approach for integrating the design requirements.

(5) Incorporate the requirements for the design, development, and test of the components into specifications and other requirements documents.

3.3.4 Systems Engineering Process.

The systems engineering process defines the system so that the resulting design will reflect the requirements and constraints related to all system elements, including equipment, computer software, facilities, procedural data, and personnel, in an integrated fashion. It is an engineering management process that is iteratively applied to the system design. Beginning at the top-level system definition, the same steps are repeated over and over until the lowest levels of the functional design are defined to the extent that the detailed design process can start. These steps are:

(1) Review the system requirements passed down from the previous design iteration. These requirements reflect the overall system level requirements that apply

to this functional element of the system as its design alternatives are analyzed during the current design iteration.

(2) Identify the next two levels of subelement components required for the design of the system. For example, for the first design iteration these two levels would be the overall system to be developed and those major subsystems (or configuration items, see paragraph 5.1) that have been identified. For the second iteration, the first level identified (and whose requirements are listed in step 1 above) is the list of major subsystems. The second level would consist of those identified components that make up each of the major subsystems from the previous level. Each iteration is continued in this manner such that the first level of system/subsystem elements identified are those elements that composed the second level from the previous iteration.

(3) For each of the subsystem (elements) listed for the second level in the preceding step, list alternative functional approaches that will meet the system requirements that have been passed down.

(4) For each of the alternatives discussed in the previous step, prepare a Requirements Allocation Sheet (RAS). These RASs list the alternatives' performance requirements and constraints in meeting these requirements.

(5) Compare the alternatives against each other and measure the ability of each to meet various attributes. These attributes will range from the operational capability to the logistic supportability and from the life cycle cost to the schedule risk associated with taking this functional approach in meeting the overall operational requirement.

(6) Choose the optimum alternative based on the above comparison.

(7) Control the requirements (RAS) associated with the alternative selected.

To facilitate this evolution of the design, the system engineering process also includes design reviews at appropriate times and at least after completion of each major phase of the development of the design. (The types, and content included, of design reviews are discussed in paragraph 6.1.) These reviews are used to ensure that the system design is given a comprehensive review at certain points in its design evolution.

The above iterative process is continued over and over until the bottom-level elements/functions are identified. Once these lower level functions are recorded, the detailed design process begins. As was the case above, the detailed design portion of the systems engineering process is also an iterative procedure. The steps involved with this detailed design are:

(1) Review the requirements (in the RAS) for the function being designed.

(2) Review the constraints introduced by lower-level detail design elements already selected.

(3) Identify all of the detail design approaches that are available for this functional element.

(4) Develop or obtain manufacturer's performance data for the design alternative.

(5) Compare the alternative detailed design approaches.

(6) Choose the optimum alternative.

(7) Record the alternative selected and place the detailed design under contractor internal control.

This procedure continues until the detailed design of each configuration item is complete. At this point, a critical design review (CDR) is conducted on each of the configuration items. These CDRs are the last point in the systems engineering

process where the Government can communicate its concern/opinion about problems with detail design elements of the CI design meeting the established program and/or stated operational requirements. At this point, the design can still be altered by the contractor (with very little increase in cost and with little delay to the schedule) before the design is released for the manufacture of hardware components or the coding of computer software programs. Depending on the certainty of the items failure to achieve the specified performance, the contractor may, or may not, accept the Government's input and make appropriate changes to the CI design. The end result of the systems engineering process is the availability of detailed engineering documentation for a proven design of system elements that will be used to build operational units. To assist the program office in successfully completing the systems engineering process, configuration management provides the technical management actions required to define, control, and track the documentation that is developed.

4. CONFIGURATION MANAGEMENT - AN OVERVIEW

As the design of the system progresses through its various stages, systems engineering is iteratively applied to flow down the operational requirements, to identify the requirements for the individual design elements, and then to define the detailed designed system to meet these requirements. To monitor this process, and to assist the program office in adequately documenting and successfully controlling the design qualified to meet the system performance requirement, configuration management (CM) is used as the technical management control system over the results of systems engineering. This section will present general information on configuration management. First, the overall function of configuration management will be defined and some of its basic concepts, objectives, and benefits discussed. Then, the role of CM in the system acquisition life cycle will be reviewed. Finally, some of the CM responsibilities of Government and contractor personnel will be outlined.

4.1 Configuration Management.

The primary purpose of CM is to assist the program manager in achieving the system performance requested by the user in the System Operational Requirements Document. Configuration management provides a means to document the design that is shown to be qualified to meet the system's operational requirements and to ensure that the system is supportable. To achieve these results, CM is applied to all the elements that compose a system - hardware, computer programs (software), support equipment, facilities, and the resulting documentation (e.g., specifications, drawings, technical orders/manuals, etc).

To accomplish this, CM focuses on major elements of the system called configuration items (CIs). A CI is any aggregate of hardware/software, or any of their discrete portions, that satisfies an end-use function and is designated by the Government for configuration management. The functions and processes involved with the CM of a system are focused on each of the system's defined CIs, which are at a fairly high level in the system design. Configuration management, however, ultimately defines and controls all levels of assembly within each CI's design. These items can be either hardware items (referred to as Configuration Items, CIs) or software items (referred to as Computer Software Configuration Items, CSCIs). [In earlier documentation, CSCIs were called Computer Program Configuration Items, CPCIs]. The selection of a subsystem, or group of subsystems, to become a configuration item should be done carefully and should be tailored for each particular program. Configuration items are those parts (items) of the system design whose performance parameters and physical characteristics must be separately defined/specified, reviewed in more detail, monitored during development, and controlled throughout the life cycle to provide management the insight needed to allow the system to achieve its overall end use function and performance capabilities. A configuration item may vary widely with respect to complexity, size, or type depending upon the system for which it is being developed, but any item that is required for logistics support or is itself designated for separate competitive procurement must be considered for selection as a CI/CSCI. Thus, the following discussions are appropriate at the system-level, but they also apply all the way down to each of the configuration items involved.

4.1.1 Formal Definition.

Configuration management provides the means for controlling and documenting the developmental process of a product as it progresses through the system acquisition life cycle. As with all military functions, CM has its formal definition stated in the military documents. According to AFR 14-1, Configuration Management, 1 December 1988, configuration management is:

> A discipline applying technical and administrative direction and surveillance to do the following:
> (a) Identify and document the functional and physical characteristics of a CI.
> (b) Control changes to the CI and its documentation.
> (c) Record and report change processing and implementation status.

Thus, CM can be looked at as maintaining technical order within the acquisition management process and is applicable to both hardware and software development. At any given time, CM should be able to supply both a current contractual description and an internally (contractor) controlled description of a developing unit of hardware, of computer software, of support equipment, of facilities, and/or of the system. It should also provide traceability from its requirements to the higher level CI's requirements and the control of changes to those requirements. Configuration management ensures repeatability in the production of the approved design of the configuration item(s), and it obtains complete information about the impacts of any changes, thereby allowing for complete analysis of the change before a decision to change is made.

This CM process should be tailored to each program based on its size, stage of acquisition life cycle, the nature and complexity of the configuration item(s) involved, and the quantity of production units being procured. CM is implemented through the application of the configuration identification process, the configuration audit (and technical review) process, the change management (both configuration and change

control) process, and the configuration status accounting process. Each of these processes, and their impact on CM, will be looked at in detail in Sections 5 through 8 of this handbook. The following subsections briefly discuss some of the basic concepts of these processes.

4.1.2 Configuration Management Objectives/Benefits.

The four CM processes (described in the following paragraphs) are used to meet the overall CM objectives for the program office. These objectives and benefits are defined in AFR 14-1 as:

(1) Assist program management in achieving, at the lowest total life cycle cost, the required system performance, a realistic schedule, and the required operational efficiency, effectiveness, logistics supportability, and readiness.

(2) Allow the maximum degree of design and development latitude while, at the same time, introducing the appropriate degree and depth of change control necessary for development, production, and logistic support.

(3) Attain maximum responsiveness and efficiency in the management of engineering changes with respect to necessity, cost, timeliness, and implementation.

(4) Provide uniformity of CM policies, procedures, documents, forms, and reports within the program office and between the program office and the contractor.

In addition to these overall program CM objectives and benefits, the CM function of the program office should ensure that:

(1) Specifications and engineering technical data are adequate for configuration needs.

(2) Configuration technical documentation is verified, up-to-date, and available.

(3) The configuration of all units and of all CIs, are known.

(4) Total performance, logistics supportability, system security and disclosure, readiness, cost, and schedule impacts of engineering change proposals (ECPs), deviations, and waivers are known at the time of approval.

(5) ECPs are processed and evaluated in a timely mannor to maintain established target schedules.

(6) Physical and functional interfaces between systems, equipment, software, and/or facilities are documented and controlled as required for the corresponding phase of the system acquisition life cycle.

(7) A comprehensive management information system is established to ensure that the status accounting process is maintained throughout the development, production, and operational phase of each of the configuration items.

(8) Performance to specified requirements and production to the documented designs can be verified.

4.1.3 Basic Concepts of CM.

Configuration management is the process by which the functional and physical (technical) characteristics of systems, hardware, software, equipment, and documentation developed, operated and supported by DOD are identified, audited and reviewed, controlled, and accounted for. The functional characteristics (configuration) refer to the performance the item is expected to achieve, while the physical characteristics (configuration) refer to the configuration item's specific design and appearance. The four processes within the overall CM function provide the means by which CM is applied to a product. Each will be briefly described in the following sections.

4.1.3.1 Basic Concepts of Configuration Identification Process. The configuration identification process involves the selection of the technical documents that describe the functional and physical characteristics of a configuration item (CI) as it progresses through its life cycle. However, care must be taken to distinguish the configuration identification process from the configuration identification. Configuration identification refers to family of technical documents (e.g., specifications, engineering drawings, software listings) which identify and/or describe the system or configuration item during the system acquisition life cycle and to the identification numbers used to distinguish the items and the documents.

The configuration identification process, and the documentation involved, become more detailed and precise as the design of the CI progresses toward production. At the completion of each phase of the system acquisition life cycle, and before the next phase is begun, the appropriate document or set of documents that comprise the CI's configuration identification are formally designated and fixed. This is called "baselining" the configuration identification. By baselining, the current identification that was formerly under the contractor's internal control is now placed under formal Governmental configuration control through authentication and contractual incorporation of the specification(s). Under most developmental programs, this process involves three baselines: functional, allocated, and product. These baselines are discussed later in paragraph 5.2. Thus, at any point in the system acquisition life cycle, the CI's current controlled configuration is defined by its baseline identification documents plus any and all approved changes to these documents.

Another important aspect of configuration identification that allows for the successful management of a system's CI/CSCIs and the corresponding technical,

baselined documents, is the use of identification numbering. This provides a way for the program to discriminate between documents (and items) with different contents and/or configurations. By using unique identification numbers, the program office is able to better control and maintain traceability on the baselined CI and the CI's approved configuration identification documents (see paragraph 5.4).

4.1.3.2 Basic Concepts of Design Reviews and Configuration Audits. The configuration audit process involves the examination and verification of the CI's conformance to its approved functional and physical characteristics. A series (one for each CI, and sometimes one on the overall system) of technically-oriented configuration audits is conducted near the conclusion of the development program to verify achievement of the specified CI requirements and to validate the documentation (by comparing the CI with its technical documentation) defining the product design.

The design reviews are an integral part of the Government oversight of the contractor's systems engineering process discussed in paragraph 3.3, The Role of Systems Engineering. These reviews occur at points during the design development when it is determined that the evolving technical design has completed a distinct stage of its development and that the results of this stage should be reviewed before the development proceeds into the next stage. The reviews are used to provide the Government the visibility and technical understanding required so that they can be assured that the system design appears to be in compliance with the specified requirements. Configuration managers become involved in the early reviews to (a) make sure that the draft documentation (normally specifications) defines the program requirements before the documentation is baselined, and (b) monitor the progression of the contractor's configuration management processes to assure that proper controls

are being invoked. At the reviews that occur later in development, configuration managers are primarily concerned about design problems which may lead to engineering changes to the baselined documents.

4.1.3.3 Basic Concepts of Change Management. The change management process involves the evaluation, coordination, decision, and implementation of all changes. This process is composed of two parts: configuration control (which is provided through the use of advanced change study notices and engineering change proposals) and change control (which is provided through the use of advanced change study notices and contract/task change proposals). Configuration control involves technical changes made to the baselined configuration of a CI. Change control involves changes to the contract that do not impact the baselined documents. Through the change management process, the program office ensures that they have all of the information required for making deliberate decisions and ensures that these changes are effectively administered and implemented.

Configuration (baseline) control is that part of the change management process by which the configuration manager (and other program office personnel) reviews a change being proposed to any of the system or CI baselines (beginning when the contractor establishes a functional baseline) to ensure that the change will be beneficial to the Government prior to that change being approved and contractually implemented. As far as the Government is concerned, only those hardware and computer software items/elements that have been baselined are subject to this type of change management (configuration control). However, the contractor will normally place any number of design documents under some type of internal control during the

development cycle far in advance of the time that the Government baselines those documents.

Change (contract) control is that part of change management through which the configuration manager (and other program office personnel) reviews any changes that are being proposed to the other contractual (non-baseline) documents in order to decide whether to authorize implementation of these changes by the contractor. This form of change management assures the program office that no changes to the contractually agreed to work tasks will be performed by the contractor without Government (program office) coordination and approval. The change proposals under this part of change management are normally oriented towards Statement of Work tasks, contract data requirements list, tailoring of management-oriented military standards for the proposed program, and contractually binding contractor plans.

The deliberate decisions made as part of the change management process are based on a determination that the proposed changes to any of the established baselines of the system's CIs (hardware or computer software), or to the contractually agreed upon work tasks, will be beneficial to the Government in terms of operational effectiveness, logistics support needs, cost, or schedule. The change administration and implementation portions of change management ensure that all approved changes to either a configuration item and its technical documentation, or to any other contractually binding non-baseline document are properly incorporated and that no other changes find their way into these documents without program office approval.

The decision making body which regulates this process for the program office is the Configuration Control Board (CCB). This is the only agency which has the official authority to act on proposed changes to the approved baseline. [NOTE: In actuality,

the CCB is not a voting organization. The CCB chairperson (normally the program manager) is responsible for making the decision after receiving inputs and recommendations from all other board members. The board members can either concur or nonconcur with the chairperson's decision. These concurrences/ nonconcurrences are recorded on the CCB Directive.] The CCB (discussed in paragraph 7.3), provides direction to the rest of the program office, and to other participating activities, for all proposed changes. Configuration management personnel have the responsibility to establish and maintain the change control procedures, to ensure that the CCB is convened to decide on each change, to formally maintain the structure of the CCB, and to record the board's decisions. Configuration managers then need to continue tracking these proposals through contract authorization and implementation to ensure timely incorporation of all approved changes (an overlap with the configuration status accounting function of configuration management).

Another form of change management that may be encountered by the program office is interface control. Interface control is primarily a requirement of the systems engineering process and is generally required when two or more contractors, or Governmental agencies, are involved in developing items whose individual configurations may affect one another. Configuration managers should be aware of the requirement for, and the application of, interface control. They should also be cognizant of the decisions being made concerning the configuration and the characteristics of an item that are being shared with, or must be designed to interface with, other items within the system's design or operational usage. Under interface control, a separate review board (called an Interface Control Working Group, which may be comprised of associate contractors and/or Government agencies) is

established to evaluate, coordinate, approve, and verify all proposed changes. However, this board must maintain a close working relationship with the primary CM groups, and particularly the CCBs, of the participating organizations, especially when changes to baselined interfaces are involved.

4.1.3.4 Basic Concepts of Configuration Status Accounting. The configuration status accounting process involves the recording and reporting of information required to effectively manage a CI's configuration, and related documentation, throughout the system acquisition life cycle. Configuration status accounting, in the form of a management information system, should provide all interested parties (implementing command, supporting command, using command, and contractors) current and historical information concerning the other three configuration management processes involved with the system's development. This information database provides the program and configuration managers the visibility and traceability of the baselined configuration items (by identification numbers and related technical documentation), the means to schedule and conduct the required technical reviews and configuration audits at the appropriate times during system development, the method to manage the processing and implementation of (approved) changes to these baselines, and the makeup of operational units delivered to the using command.

The information that is available through status accounting gives the program office, and other participating agencies, personnel the means to coordinate the actions that must be performed to successfully accomplish the systems engineering/design process, item manufacture, system deployment, and logistics support of the delivered operational units. It will also allow them to monitor the changes to the system CIs as they arise throughout the system acquisition life cycle. In addition, this information

provides program management with the means to determine if changes are being implemented as directed.

This CM process (configuration status accounting) is normally not used during the concept exploration/definition phase of the system acquisition life cycle because piece parts, documents, and/or CIs are not yet completely defined. During concept demonstration/validation the program office and contractor will normally agree upon the system specification and hence a functional baseline. At this point in the program's development configuration status accounting becomes a necessity to track and record the specification and the changes to the functional baseline. During full-scale development and prior to the establishment of a product baseline, the configuration status accounting information provided by the contractor (and program office in some instances), is used to define (and trace) the current (and historical) status of the configuration item specifications and to identify current status (e.g., in review, disapproved, approved waiting implementation, or implemented in all (some) of the affected items) of change proposals received by the program office.

After the establishment of the product baseline, the information recorded, stored, and reported through configuration status accounting is greatly expanded to address many elements of the entire system. In addition to the expansion to include the same functions mentioned above for the product baseline documentation, the CM process is used to document the status (current and historical) of the configuration of every operational unit delivered to the Government and to trace any approved change to these units to ensure its implementation (either through retrofit, repair and replace, or modification actions).

4.1.3.5 Summary.

The four processes which comprise configuration management are used by the program office as management/control mechanisms to ensure that the systems engineering process is successful in developing a system that meets the stated requirements. These processes identify, review and audit, control, and account for the design of the system's CIs. The configuration identification process answers the question "What is the system configuration?" by providing the program office the necessary technical documentation that defines the parts of the system. The design review and configuration audit process furnishes the program office the answer to the question "Does the system being built by the contractor satisfy the stated needs and/or requirements?" by providing the means by which the system design may be compared to the system/item specifications. The change management process provides the program office the answer to the question "How do we control changes to the configuration of the system and items being developed?" by providing the necessary information required to process changes that directly or indirectly impact either the technical or contractual documentation. Finally, the configuration status accounting process is used to answer the question "What changes have been made to the system?" by documenting the configuration of the baselined design and documentation and maintaining a listing of the approved changes (including those already implemented or those pending implementation with a schedule for implementation) to these baselines.

4.2 CM and the System Acquisition Life Cycle.

The activities involved with each of the configuration management processes should be consistent with the objectives of the program/product and be appropriate for

the system acquisition life cycle phase. CM is required in varying degrees throughout the entire system acquisition life cycle, generally beginning with the control of the top level requirements and increasing in scope as the product design evolves, to ultimately controlling the detailed system design.

4.2.1 The Role of CM in the System Acquisition Life Cycle.

The CM functions associated with configuration identification, configuration audits, configuration control, and configuration status accounting and their relationship to the system acquisition life cycle of a major weapon system, are shown in Figure 3. [Note: The detailed timing of, requirements for establishment of each baseline, and the documentation involved will be discussed later in Section 5.] It is important to note that this is only a general picture of the typical timing; the actual timing for any particular major system are dependent upon that program's schedule and complexity. The relationships shown constitute those which would be viewed as the ideal case, where each baseline is established and documented before the start of the next life cycle phase. However, because of program uncertainues, decision making requirements, and the inherent instability of the development process, establishment of the baselines for some of the CIs is sometimes delayed. Yet, the relationships shown reflect the baselining policy established in AFR 14-1 to which most programs should try to adhere. [Note: It should also be pointed out that the hardware and computer software CI development cycles are, in most cases, parallel series of activities performed as part of the system acquisition life cycle. While each progresses at its own schedule, each CI design decision must consider the interface characteristics being shared with other CIs. As such, not all CIs for any single program will necessarily meet these baseline timing and resulting CM activities as shown.]

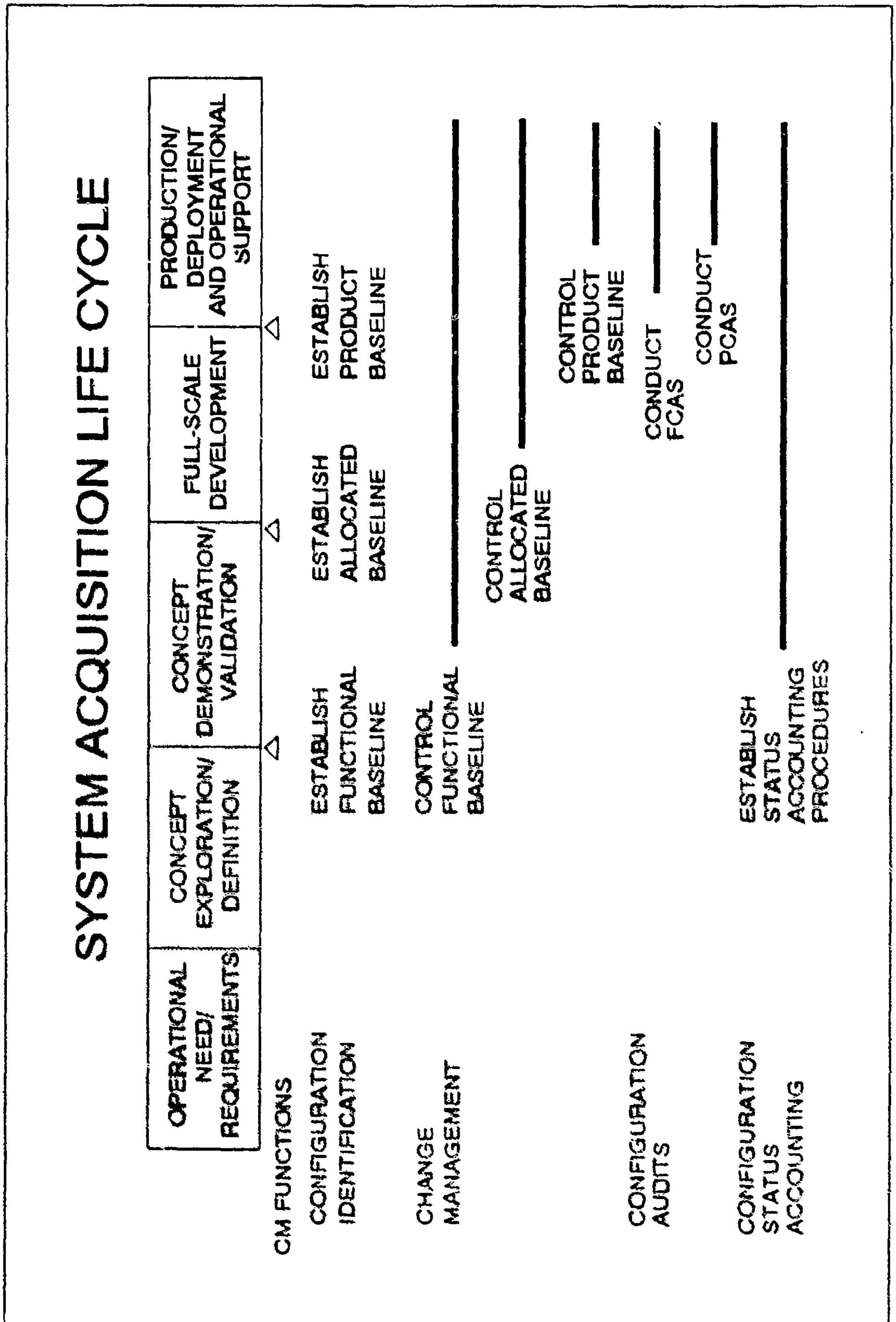

Figure 3: CM and the System Acquisition Life Cycle

The following subsections discuss the role of CM with regards to both hardware and software CIs during the system acquisition cycle. Included at the end of each acquisition phase is a list of possible tasks that may be performed during that phase that may involve CM personnel. The various tasks will not be defined in detail. Each of these tasks will be discussed in Sections 5 through 8, as applicable. The intention here is to give the reader an idea of what types of CM-oriented tasks may be in work during the individual system acquisition life cycle phases.

4.2.1.1 Concept Exploration/Definition Phase. During this phase, the program office is attempting to define the overall mission and system requirements and to explore and prioritize a multitude of alternative approaches. The major product is the preliminary System Specification and would also include System Segment Specifications, if required. If any prototype hardware or software component is developed for data gathering purposes, it usually does not need to be defined in a separate specification or subjected to configuration management at this time. However, it may be advisable to obtain design disclosure documentation for the prototype.

Normally, CM is not required during this phase except for the identification of the draft system-level functional requirements in the draft version of the System/System Segment Specification. As a part of the development of the system specification, one or more System Requirements Reviews (SRRs) may be conducted. If possible, this phase will end with the approval of the System/System Segment Specification and the establishment of the functional baseline.

There will be some programs where the system design process may progress further. In these instances the functional baseline may be ready to be established prior to the completion of this first phase in the system acquisition life cycle. If this is

the case, the program office may also want to establish a Configuration (Change) Control Board to maintain change management over the functional baseline. In addition, the program office may have to request that the contractor review any Interface Control Drawings/Documents prepared to identify the system/system segment interfaces with other system/system segment(s).

4.2.1.2 Concept Demonstration/Validation Phase. The purpose of this phase is to validate the remaining system concepts and refine the basic characteristics of the system. Based on the approved acquisition strategy, the program office investigate several alternative system concepts or may only be allowed to investigate subsystem alternatives. The primary CM documentation generated is an updated System/System Segment Specification and preliminary hardware CI Development (and computer software CI Requirements) Specifications. While not CM documentation, an initial version of the Computer Resources Lifecycle Management Plan will also be generated. Configuration management will control and account for system-level, and identify the draft CI-level, functional and interface characteristics. Some of the CM tasks that may be accomplished during this phase are:

(1) If the functional baseline was not established during the previous stage, it should be established sometime during this phase.

(2) After the System/System Segment Specifications have been baselined, and the functional baseline has been placed under contractual control, configuration control procedures for processing, approving or disapproving, and implementing approved engineering change proposals must be established.

(3) Once the functional baseline has been established, a configuration status accounting database must be established that is able to begin tracing any activities relating to the System/System Segment Specifications.

(4) One or more SRRs may be conducted, and System Design Reviews (SDRs) for hardware CIs and Software Specification Reviews (SSRs) for computer software CIs may also be conducted (see paragraphs 6.1.1, 6.1.2, and 6.1.3).

(5) As the system requirements and design constraints become better defined, then the identification/selection of hardware and computer software CIs must be accomplished. (This task will probably have been started during the last phase.)

(6) Prepare (normally by the contractor) the draft hardware CI development and computer software CI requirement specifications.

(8) Review the contractor's CM documentation to include CM Plans describing the actions required for each hardware and computer software CI, the Computer Resources Life Cycle Management Plan, and any draft test plans for the CI/CSCIs.

(9) If a SDR is conducted for a CI, the program office may also choose to authenticate the development specification and establish the allocated baselines for the CI. If possible, the development/requirement specifications will be authenticated and the allocated baselines will be established for at least the top-level CI/CSCIs by the end of this phase.

4.2.1.3 Full-Scale Development (FSD) Phase. During FSD, system hardware and software CI prototypes are designed, built, and tested. These CIs are then integrated into the complete system and tested, as a system under conditions as close to operational as are achievable at the test site(s). CM documentation during this phase will be a final hardware or computer software CI development or requirement

specification and a draft product specification for each CI/CSCI. Other program documentation that may be produced during this stage, that should be of concern to the configuration manager, are the revised Computer Resources Lifecycle Management Plan and the CI test plans. In addition, prototype CI/CSCIs are usually produced during this phase. If so, the configuration manager should evaluate/monitor the contractor's system for tracking the detail design of the developed prototype to assure that it is accurately recorded in the internal design documentation. Thus, CM will control, account for, and audit system and CI-level functional and interface characteristics, as well as, begin to identify draft CI-level detailed design characteristics. CM tasks that may be performed include:

(1) Authenticate, if not accomplished last phase, the Development Specifications for hardware CIs and the Requirement Specifications for CSCIs. Also, establish the corresponding allocated baselines for these CI/CSCIs.

(2) Continue configuration status accounting of the functional baseline, and begin the status accounting process on the established allocated baseline(s). In addition, those change management procedures that were established to control the functional baseline are now expanded to be able to also control any proposed changes to the allocated baseline(s).

(3) Attend Preliminary and Critical Design Reviews.

(4) Review updates to the contractor's CM documents (e.g., CM Plan) that were initially submitted during concept demonstration/validation.

(5) Begin the review of the CI Product Specifications. If desired, the Product Specifications can be authenticated (most likely with CSCIs) and the CI's detail design baselined upon successful completion of a physical configuration audit.

(6) Conduct functional configuration audits on each configuration item. Each audit may be for only one CI, or it may be conducted for a group of CIs. However, all CIs must be audited. For major weapon systems requiring integrated system-level testing, the program office may also want to conduct a functional system audit.

(7) If the present contractor has not already been identified/selected/approved to continue the program into the production/deployment and operational support phase of the system acquisition life cycle, then the configuration manager must ensure that physical configuration audits are also performed on the developed CI/CSCIs (usually prototype units). In addition, if this requires any CI/CSCIs to be delivered to the program office, the configuration manager must be able to validate and verify that the design of the delivered item matches the current approved, baselined CI/CSCI design.

4.2.1.4 Production/Deployment and Operational Support Phase. CM will audit, control, and account for the CI-level detail design characteristics. In addition, CM will control and account for the actual configuration of units in the inventory. CM tasks during this phase include:

(1) Authenticate and baseline, if not accomplished during the full-scale development phase, the product specifications and establish the resulting product baselines.

(2) Implement logistics support system and inventory tracking status accounting procedures while continuing the status accounting started during full-scale development.

(3) For most major weapon systems, when the contractor has already been pre-selected to perform the program's production phase prior to the completion of full-scale

development, the physical configuration audits are performed on each CI/CSCI during the production/deployment sub-phase of this acquisition life cycle phase.

(4) The implementing and supporting commands need to prepare for the transfer of CM responsibility from the program office to AFLC and the appropriate Air Logistics Center. This will occur when system PMRT is conducted.

(5) The supporting command must be ready to accept responsibility for, and have procedures in place to accept, the contractor's status accounting information. The supporting command must be able to trace the current documentation/identification numbers, the status of changes, and the configuration of the operational units including changes due to retrofit, repair and replace, and modification actions.

4.3 Configuration Management Responsibilities.

To facilitate successful CM of a major system throughout the acquisition life cycle, both the Government and the contractor must be able to perform certain CM responsibilities.

4.3.1 Government Responsibilities.

The CM responsibilities required of the Government are accomplished, depending on where the program is in the system acquisition life cycle, by the implementing command and/or the supporting command. The implementing command (normally AFSC) transfers the CM responsibility of the program to the supporting command (normally AFLC), along with all other program responsibilities, at system PMRT. Besides being responsible for, and providing CM support after system PMRT, the supporting command should also be active early in the program's development by

providing inputs into the initial CM plans and requirements and by helping monitor the selection and development of CIs during system design.

The implementing command is responsible for ensuring that the CM tasks are accomplished throughout the development and testing of both hardware and computer software CIs. The individuals in the program office that are held responsible for a successful CM program are the program manager and the configuration management directorate personnel.

4.3.1.1 Program Manager CM Responsibilities. The program manager is ultimately responsible for establishing and implementing a CM program that identifies, documents, and controls the functional and physical characteristics of all hardware and software CIs. The CM directorate usually provides the functional inputs that allow the program manager to make reasonable and deliberate CM decisions.

4.3.1.2 Configuration Management Directorate Responsibilities. The CM Directorate of the program office is responsible for establishing and implementing the detailed policies and procedures for all CIs under the Program Manager's direction. Some specific responsibilities are:

(1) Provide CM tasks and data item requirements for incorporation into the Request for Proposals (e.g., Statement of Work paragraphs and CDRL forms) and contracts.

(2) Coordinate CM requirements with the using and supporting activities.

(3) Review, and approve, contractor developed and contractually required to be delivered CM Plans. In addition, they should review the internal plans and procedures

of the contractor, as they relate to CM, to insure that the contractor has the ability to accomplish a successful CM program.

(4) Monitor contractor implementation of contractual CM requirements.

(5) Ensure that functional, allocated, and product configuration identifications of system hardware and computer software CIs are correctly documented and baselined.

(6) Act as the focal point within the program office for centralized specification and CI control.

(7) Receive, review, process, and distribute change management proposals (both engineering changes and contract changes) to the affected agencies and contractors.

(8) Establish a change management/status accounting procedure that is capable of monitoring the contractor's efforts in the implementation and incorporation of approved changes into the established baseline documents, into the corresponding configuration item identification, and into the system elements (e.g., manuals, support software) affected by the change.

(9) Manage the process that controls engineering changes to baselined documents and CIs.

(10) Maintain currency of CCB orders.

(11) Provide a secretariat, for the program office CCB, who is responsible for recording and reporting all results.

(12) Ensure that the system level configuration status accounting information is maintained and that the information is readily available in an understandable format.

(13) Plan and conduct configuration audits jointly with the contractor(s).

(14) Prepare the CM portion of the PMRT package for transfer to the supporting command.

4.3.2 Contractor Responsibilities.

The contractor CM responsibilities are tailored to each contract. They will vary a great deal depending on the size and complexity of the development task. Some of the basic CM tasks imposed on a contractor include:

(1) Preparing internal CM directive, procedures, and plans that document their responsibilities and procedures for implementing CM on both hardware and computer software CIs. Depending on the size of the program (and almost always required for major weapon systems) the contractor may be required to submit its CM Plan to the Government for concurrence/approval.

(2) Document the design considerations for each CI/CSCI in its development, requirements, and product specification. With these specifications, define and establish the appropriate baselines (i.e., allocated and product).

(3) Assign or obtain identification numbers (see Section 5.3) for all CIs, the documentation, and appropriate component parts.

(4) Maintain the master copies of all specifications, drawings, computer software listings, and other technical documents required to identify the system and the CI/CSCIs.

(5) Develop a process by which required change management proposal documents are adequately coordinated to assure completeness of the content prior to their release to the Government. If the proposed change is accepted by the Government, then the contractor's process must be ready to trace the approved change through its implementation and incorporation into the baselined technical documents (e.g., drawings, specifications, computer software listings) and other affected system elements.

(6) Establish an internal configuration control system for non-baselined documents, hardware, and computer software during the development and qualification testing of each CI/CSCI.

(7) Develop a configuration status accounting process (usually in the form of a management information system) that is capable of recording, maintaining, and reporting the outputs associated with the other three CM functions.

(8) Establish, if required, interface control working relationships with other contractors participating in the development of the system or subsystems.

(9) Prepare for, and help the program office conduct, functional and physical configuration audits for each CI/CSCI of the system. In addition, for major weapon systems if there are system-level requirements that can not be verified after the CI-level functional configuration audits are completed, then the contractor may also be requested to prepare and help conduct a functional system audit.

(10) Require subcontractors and vendors to plan and implement CM measures and monitor the implementation of these plans and measures. Also, conduct any necessary audits of subcontractors and/or vendor supplied items, and provide a change process to maintain control of these developing designs.

5. CONFIGURATION IDENTIFICATION

The underlying requirements for the application of configuration management by the program office are: (1) to adequately define/identify the system design so that it addresses both the performance and the logistics support of the system/items being developed, and (2) to facilitate the continued logistics supportability of the system/items in operational use as changes are made throughout their operational life. As such, the system and its parts must be identified and defined in both functional and physical terms. The CM process that performs these tasks of identifying and defining the system and its parts is called configuration identification.

Configuration identification provides the means by which the performance, qualification, fabrication, and acceptance requirements associated with the product under development are progressively defined, documented, and placed under control. Using the appropriate technical documents (including specifications, drawings, computer software listings, and parts lists), the configuration identification process establishes baselines which contractually define progressively more detailed descriptions of the functional and physical characteristics of the items being bought. These CM activities are accomplished against the overall system and against those parts of the system which have been identified as configuration items. The configuration identification process also helps the program manager to measure the contractor's progress by providing the basis for comparing the evolving design to the stated operational requirements of the desired system/item which were previously stated by the Government. The contractual baselines (discussed in paragraph 5.2) are established at points in the program where it is deemed necessary to provide a definable and manageable departure point for the development and production of the

system or item. The baseline, plus any approved changes to it, constitutes the current contractually binding configuration identification (that is the technical definition of what the Government expects the system/item to accomplish).

To provide further insight into the CM process of configuration identification, this section will begin by discussing the role of configuration items (CIs) within the systems engineering/design processes, criteria for their selection, and the results the program office can obtain from their CIs. Then, the concept of baseline management, and the various baselines used, will be discussed to show how they relate to configuration identification. Next, the various types of technical documentation that may be encountered during a program's acquisition life cycle will be examined. Finally, after the documents have been identified, this section will outline the idea of identification numbering and how the various items and documents are numbered for simpler identification.

5.1 Configuration Items.

As the development of the system proceeds, and the design is iterated, a better understanding of the system and its functions evolves, resulting in the breaking down of the stated operational requirements into functions that can be assigned to system elements. As a part of the systems engineering process begun in the concept exploration/definition phase of the system acquisition life cycle, elements of the proposed system are identified, requirements (or functions) are allocated to these elements, and trade-off analyses are performed to optimize the system functional approach as a whole. Figure 4 shows a sample of a functional breakdown (not to be considered all inclusive) of a product into various levels:

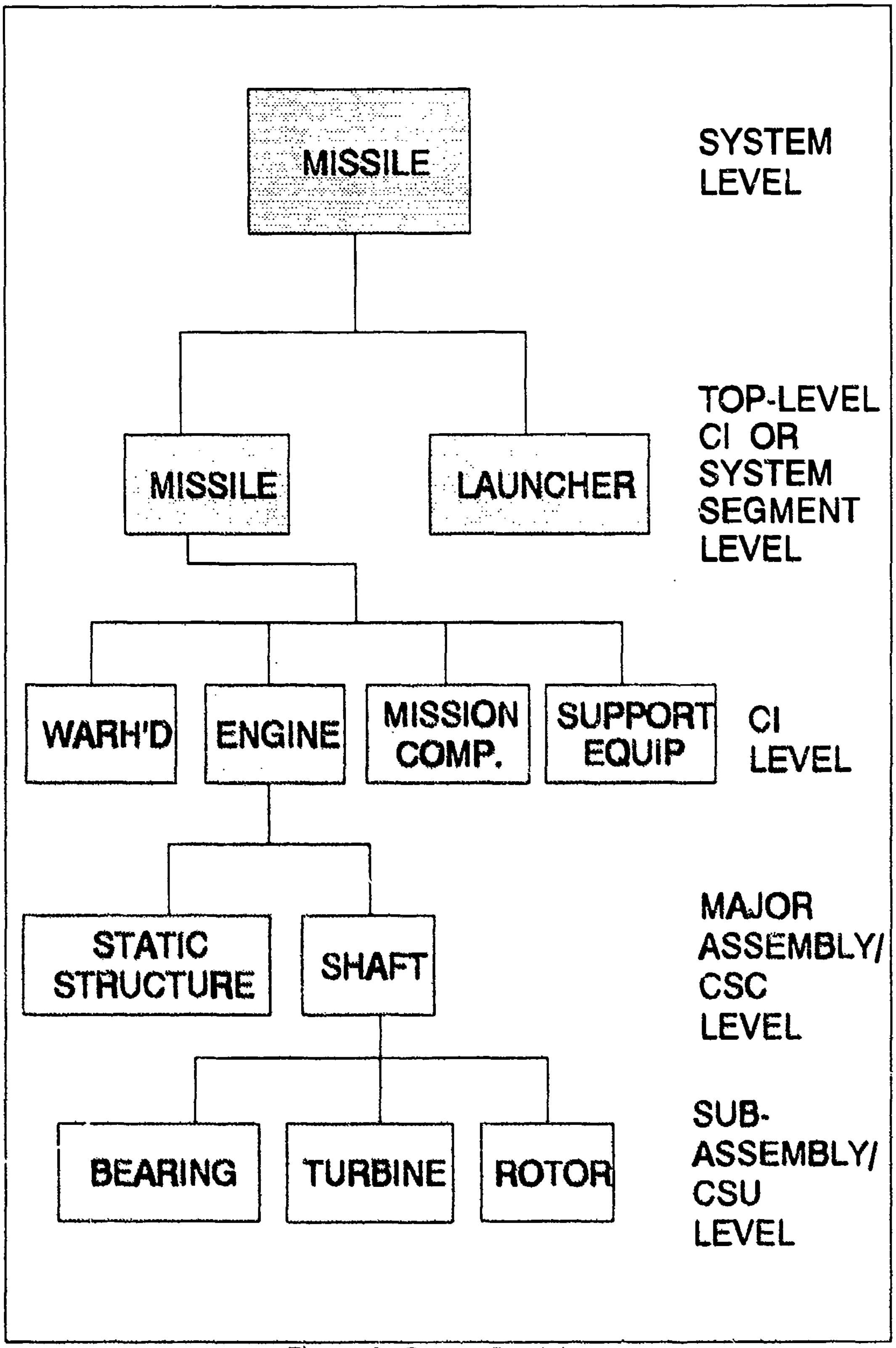

Figure 4: System Breakdown

1. System Level: This is a composite of all the items (e.g., air-to-air missile or air-to-ground missile), assemblies, and support elements capable of performing or supporting the operational requirement. The complete system includes the equipment. computer software, related facilities, material, supply system, trainers, and personnel required for its operation. The system is considered a self-sufficient item in its intended operational and/or support environment. This system-level item is defined to the program office by HQ USAF through its issuance of the Program Management Directive. Development of a system may be accomplished by a single contractor or it may be accomplished by several contractors working on separate portions of the system.

2. Top-Level CI or System Segment Level: These top-level functional elements of the system are discrete packages of system performance requirements, functional interfaces, and configuration items separately contracted to one contractor or assigned to one Government organization, which is then responsible to the procuring activity for that segment of the system's performance. This level may be used when the development of major portions of the system (which are essentially systems in themselves) are separately contracted by the program office with more than one contractor. In addition, this level may be used when a system is acquired on an incremental or evolutionary basis or when an existing system is to be given a major additional capability via a modification. In most cases, however, the system segment (and the related System Segment Specification) is not necessary; these top-level functions would normally be documented as CIs in Prime Item (see paragraph 5.3.1.2.1) Specifications. The system segment may consist of one or more subsystems. For example, as shown in Figure 4, if the program office were developing

a surface-to-air missile, then the system might be divided into two system segments. One segment might be the missile and the other the launcher system.

3. Configuration Item Level: This level (which is often several levels) of the system breakdown includes all hardware and computer software functional elements and subelements which are identified as requiring additional developmental of logistics definition and visibility, that is, as CIs. These CIs should be capable of some independent operation that satisfies an end use function. The hardware CIs include those visible parts of the system required to accomplish or support the mission (e.g., the engine, support equipment, mission computer, and warhead), while the computer software units would be those instructions and definitions required for the computer to perform some computational or control function to accomplish or support the mission. For a system under development, the contractor (system engineering) recommends elements as CIs; the program office is responsible for final selection of the CIs (both hardware and computer software) for the program. For each configuration item, there may be two additional levels of assembly which are encountered.

4a. Major Assembly Level: This level is comprised of units (each of which is comprised of a number of parts and subassemblies) that when joined together perform a specific function. It should be pointed out that, for any system, a CI may also be a major assembly for some other CI (and the same is true for CSCIs). That is, any CI/CSCI may be a subelement of some other higher-level CI/CSCI. This level of assembly is defined by the contractor and presented to the program office for their concurrence. In the example of Figure 4, the configuration item-level engine is separated into two (probably more in a true system breakdown) major assemblies: the shaft (all moving parts) and the static structure (all non-moving parts).

4b. Computer Software Component (CSC) Level: These are functionally or logically distinct parts of a CSCI that are distinguished as separate for purposes of convenience in designing and specifying a complex CSCI as an assembly of subordinate elements or subprograms. These parts of the CSCI are defined and designated by the contractor and may be composed by other CSCs or computer software units.

5a. Subassembly Level: Items at this level are made up by two or more parts (pieces joined together which can not normally be disassembled without their destruction), which form a replaceable portion of the major assembly. These subassemblies are defined by the contractor and presented to the program office for concurrence. In Figure 4, the subassembly items of the shaft are the bearing, turbine, and rotor.

5b. Computer Software Unit (CSU) Level: This is the lowest compilable element of a computer software program. CSUs are the actual physical entities implemented in code and are separately testable. The CSUs are also defined and designated by the contractor.

Remember, the breakdown shown is only an example of how this particular system/program could be partitioned. Any other program could be broken down with completely different results depending on the technological or other programmatic risks that may have to be addressed. For those "advanced" technological programs, there may be other levels inserted between those levels shown in Figure 4. However, it is important to remember that CM activities are accomplished at the CI level.

5.1.1 Concept of CIs.

CIs are those parts of the system on which the principles of CM are applied. That is, for each CI: (1) there will be a separate development (requirements) and product specification generated, (2) there will be separate design reviews and configuration audits, (3) for engineering changes which affect more than one CI, a separate Engineering Change Proposal (ECP) package is required for each CI affected, and (4) status accounting information is tracked against each CI. Regardless of program size, for each new hardware or computer software item being developed under a separate contract, there must be at least one CI designated. Major functional elements of this new design that incorporate a new technology, or are critical to the successful performance of the overall system, are generally designated as a CIs.

As a part of the systems acquisition process, the segmenting of the system into CIs enhances the effectiveness of the monitoring of critical hardware/software item development, allows the program office to separately define performance and test requirements for these significant items, and provides for a manageable span of technical control. When the system enters its operational phase, the documentation and records kept for the CIs during their development provides the basis for adequate and effective management and support of the system. Adequate quality documentation will be available to allow for evaluation of proposed changes to system capabilities; it will provide sufficient information to allow for competitive reprocurement of needed parts, thereby ensuring lower cost of logistics support; and it will allow the supporting and using agencies to maintain traceability of all critical pieces of that system. However, all these advantages obtained through the use of CIs are offset by the cost associated with the development of this documentation for each configuration

item and the cost of the management activities for each CI. Therefore, only those system elements which really need this additional documentation and management emphasis should be selected as CIs.

5.1.2 Selection of CIs.

The selection of a hardware or computer software element to be managed as a configuration item is governed by the requirement for separate control (by the Government) of the item's inherent characteristics or control of its interfaces with any other configuration item. If the CI's inherent characteristics and interfaces are not separately controlled with its own documentation, these same characteristics and interfaces are still defined and controlled in a less definitive sense at the next higher-level CI or system. The selection is also dependent on the need to provide more management effort toward, and more detailed visibility into, the design for a specific item. The initial selection of those items recommended to be considered as CIs is most often performed by the contractor. However, the final approval is usually performed by the program manager using inputs from the engineers, project manager, configuration manager, and a representative of the AFLC System Manager. The program manager must weigh the benefits obtained from selecting an item as a separate CI versus the additional costs associated with obtaining the separate documentation and control over the item of interest.

When deciding whether an item should, or should not, be included as a CI during system design, the program office must consider the use of the item during both its development and its later operation. From a developmental standpoint, the status of being a CI would ensure that the performance capabilities of the item were monitored and reviewed during its design; the selection provides technical definition (contract)

and managerial oversight (verification) that the program office needs in developing the system as CIs. Technical considerations would include determining if the item is being developed using a new risky technology; if the failure of the item in operation would significantly impact system security or the accomplishment of the mission; or if there were a risk of performance failure. The managerial considerations would include determining if there are technically and functionally (i.e. training, mission, test) dissimilar elements of the system which could be better managed as a separate entity; whether the system or items are so large and complex that it is better to separate them into separate entities of more manageable proportions; if there is a schedule risk (the likelihood of major slips in the program schedule due to developmental problems) associated with a particular item of the system; or if problems during development could cause significant financial impacts/overruns on the program. These technical and managerial considerations, if applied judiciously, will provide a manageable span of technical control on the system and its parts during development while minimizing the costs of the documentation and management activities.

When trying to identify those elements which should be broken out as separate CIs late in full-scale development or in the production phase, there are support and competitive reprocurement considerations that the program office should weigh in selecting subelements of the system as CIs. Support considerations would include determining whether: (1) different agencies (or different locations) have responsibility for maintaining major elements of the system which have not already been documented as CIs and the need for separable documentation to manage them, or (2) whether the support concept being used for the item requires separate documentation to be available (primarily for computer software) so that it can be managed by a

designated site, Air Logistics Center, or operating command. The judicious selection of additional elements as CIs can significantly lower the cost of its logistics support and simplify the traceability of critical components. The final area that the program office should weigh while selecting subelements of the system as CIs is whether the item will be competitively reprocured. Unlike the first three areas of consideration, this last area should not, by itself, identify items as CIs. However, if the item is considered complex and is also known (or later determined) to have to undergo competitive reprocurement, then it may be selected as a CI. Selection of an item as a CI for competitive reprocurement ensures that there will be adequate separate documentation to define the item's performance requirements and design detail and acceptance tests to verify and validate the item produced and delivered.

5.1.2.1 Selecting Too Many CIs. If the program selects too many CIs for the product under development, the visibility and management will be hampered rather than improved. There will be an increased administrative burden in the engineering change process, since a separate ECP package is required for each CI affected by the change. Developmental time is likely to increase, and the costs associated with the specifications (and related documentation) and with the design reviews and configuration audits are likely to "skyrocket". Finally, maintaining coordination of all the paper work being generated will be extremely difficult. With so many CIs, there will be an increase in the complexity of the interface requirements which must be coordinated among the different items under development.

5.1.2.2 Selecting Too Few CIs. On the other hand, if the program does not select enough CIs, there may be difficulties both with monitoring the development of the

system and with system logistics and maintenance. During development, specifications for these very large CIs will be quite voluminous and difficult to work with (e.g., How much of the reliability requirement applies to this CI element?), and the system technical reviews will be muddled affairs addressing large numbers of unrelated functional elements unless they are performed incrementally (on CI candidates).

During deployment and operational support, maintenance personnel will have more difficulty controlling the design and tracking the location of individual remove or replace items if these items are documented and controlled as a part of a larger subsystem-level CI. Because of this, the identity of that portion of the larger subsystem-level CI affected by the maintenance actions may not be traceable.

5.1.2.3 CI Selection Checklist. The program manager, configuration manager, and system engineers need to insure that they have enough CIs to facilitate the development, production, and support of a system without selecting too many. As can be seen by the previous two paragraphs, that selection of the correct number of CIs is a not easy. One way to assist these managers is to use the following questions (obtained from MIL-STD-483). If the answers to two of these questions are NO, then the item under consideration should probably not be a CI. If on the other hand, most of the answers are YES, then the item should probably be considered/selected as a CI. However, in all cases the program and configuration managers must use good judgement, often basing their decision on other factors such as their past experience. These questions are only one tool to help make CI selections. Even though they may indicate a NO decision for a particular element of the overall system, some other

consideration may dictate that it should be a CI anyway. However, the following questions provide a good starting point:

1. Is the item a safety concern, and/or is it a critical high risk item?

2. For hardware items, is it readily identifiable with respect to size, shape, and weight?

3. Is the design of the item (hardware or software) being newly developed?

4. Is the design of the item incorporating new technology?

5. Will the item have any interface with either hardware or software being developed under another contract?

6. Will the item interface, with respect to form, fit, or function, with other CIs whose configuration is being controlled by other Government or contractor entities?

7. Will there be a requirement to know the exact configuration and the status of any changes to the item during its progression through the system life cycle?

5.1.3 Benefits of CI Selection.

Once the decision has been made to select certain items as CIs, what benefits can the program office expect to see from the increased cost, schedule, and performance required of the item's contractor? Basically, the benefits of making an item a CI are found in the separate specifications (focused requirements documentation) and in the progression of technical reviews and management events that are required for each CI. While separate documentation (primarily specifications) is required for all CIs, the technical reviews and configuration audits can sometimes be tailored for a CI to combine it with the reviews and audits for other CIs, thus saving the program office some cost, schedule, and manpower requirements. If such tailoring is desirable, it

should be defined in a contractual document (e.g., Statement of Work, Configuration Management Plan, etc., [see Section 9]).

5.1.3.1 Documentation Benefits. The following are some of the documentation benefits that may be realized with the selection of an item as a CI.

1. Formal preparation of a discrete [hardware CI] development or [computer software CI] requirements specification and a companion product specification. A discrete package of detail design documentation for the CI (including drawings, source codes, or parts lists).

2. Discrete configuration item identifiers (including a nomenclature), separate nameplates, and separate identification indexes to trace the history of the configuration item identification.

3. Separate ECP documents for proposed changes affecting each CI.

4. Preparation of separate operator and user manuals.

5. Accurate recording/tracking of status accounting information for each of the CI.

5.1.3.2 Management Event Benefits.

The following are some of the management event benefits that may be realized with the selection of an item as a CI.

1. Individual item qualification testing, design review activities during development, and physical and functional audits at the conclusion of development.

2. Providing traceability of detailed design.

3. Separate ECP preparation, review, disapproval or approval, and negotiation form implementation and incorporation.

4. Government change control decisions concerning the baselined configuration identification of the CI.

5.2 Baseline Management.

With items identified as CIs, the principles of CM can be applied. These principles are applied throughout the system development cycle at the CI level. Figure 5 shows the activities involved in the configuration management processes of configuration identification, technical reviews, and configuration audits (the reviews and audits are discussed in Section 6) as the system undergoes its development. Normally after completion of an appropriate technical review or configuration audit, the appropriate configuration identification documents for a CI or system are baselined to provide a full, contractually-binding technical description of the functional and physical characteristics of the item. After these documents have been baselined, all approved changes to them will require full documentation (in the form of engineering change proposals [see paragraph 7.1.1.2]), program office approval, and contractual incorporation. Thus, baseline management is an important aspect of the configuration identification process and is central to the configuration management process. It provides the "tie" between the configuration identification and change management (configuration control) processes.

Baseline management ensures the timely incorporation of the appropriate kind and detail of requirements into the contract to provide a record of the contractor's progress, a departure point for further design or development, and a basis for determining the contractor's compliance with Government requirements. As shown in Figure 5, for all large programs, three baselines are normally employed during the system acquisition life cycle. These three baselines are called the functional, allocated, and product

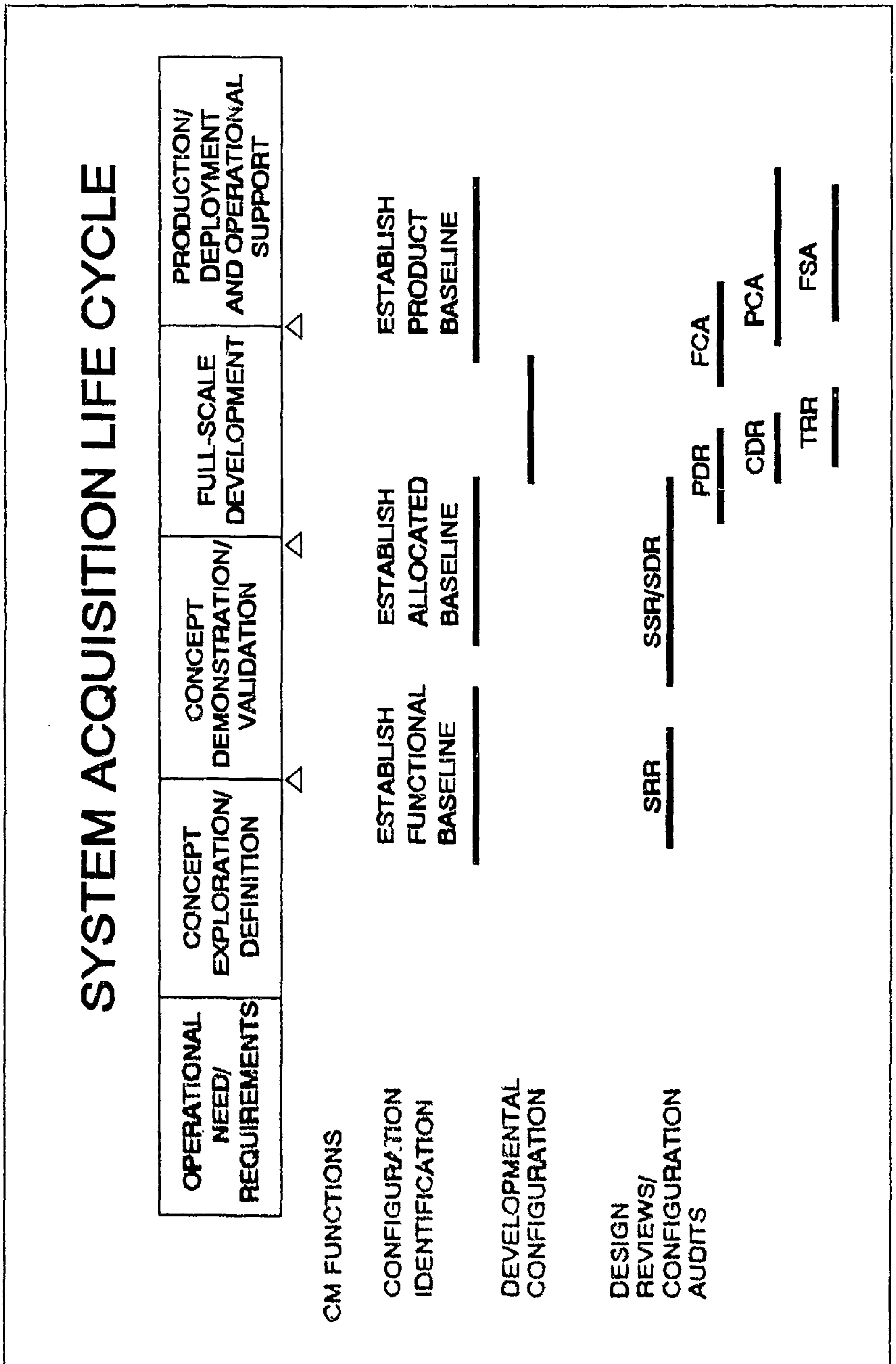

Figure 5: Configuration Management By Baseline Management

baselines. Briefly, the functional baseline documents the system performance and design requirements; the allocated baseline documents the flow down of these performance and design requirements to each CI; and the product baseline documents the detailed design to which each CI must be built. While there are normal times in the system acquisition life cycle when the baselines are expected to be established, the baselines may be established at any point in the system development process where it becomes necessary to define a formal departure point for control of future changes in both the performance and the detail design of the CI involved. As the contractor progresses with the design and development of each CI, the baselines are used to provide an increasing detailed contractual definition of the performance and of the detail design of the system hardware and computer software items.

A major assumption of the baseline management concept is that the contractor and Government can agree on an initial statement of performance requirements. Once this initial baseline is established, any changes to these requirements must be documented, priced, reviewed, and, if deemed appropriate, approved.

Since configuration management is oriented towards the concept of change management, the use of three separate baselines during the design process allows the contractor, during the early stages of development, to have considerable flexibility to make detail design changes without Government approval. As development progresses, some of this detail design flexibility is lost until, after successful completion of the physical configuration audit, the product (detail design) baseline is established and all detail design changes require Government approval. This is accomplished by ensuring that the item is documented in progressively greater detail as it progresses through development. After it has been baselined, the contractor may investigate

changes to the baselined design until all related impacts have been identified and documented. At that point, the contractor provides the change documentation which reports these changes to the program office for approval. If approved, the changes are incorporated to update the configuration identification. Thus at any point, the approved (contractually binding) configuration of an item is identified in its baseline and all approved changes from that baseline.

Due to the fact that, during its development, a CSCI is easily changed, the contractor will normally employ a Developmental Configuration, after the CSCI allocated baseline is established to internally control the CSCI design documentation as it is developed. This "pseudo" fourth baseline (see Figure 5), is internally established by the contractor on each CSCI and is used to describe the evolving configuration of the CSCI design as it is developed, and to maintain internal control of the CSCI design as it progresses between its allocated and product baselines. The program office's configuration manager and systems engineers need to be aware of the contractor's Developmental Configuration and use it to monitor the evolving design of the CSCI.

5.2.1 Functional Baseline.

This is the first baseline established during the development and acquisition of a system. The functional baseline is required for all systems and defines the functional characteristics and related verification requirements for the system. The functional characteristics defined include the overall performance, interface, and operational requirements; the design constraints; the personnel and training constraints; and the logistics support constraints for the system. These functional requirements and constraints, along with a few detailed physical interfaces, are usually contained in a

single system specification that comprises the contractual technical definition of the system capabilities. As noted in Figure 5, the authentication (approval) of this Type A (System/System Segment) Specification constitutes the establishment of the functional baseline and should normally occur early in the concept demonstration/validation phase after the System Requirements Review and not later than the completion of the System Design Review. Once agreed upon, the functional baseline provides the contractually binding technical basis for the development process of the system. When these system level requirements have been coordinated and approved, the program may then proceed to identify the critical elements (CIs) which will then be identified in separate, lower-level specifications to technically define the next baseline.

5.2.1.1 Functional Configuration Identification. The definition of a baseline characterizes it as an unchanging picture of the contractual technical requirements when it was established; the current "baseline" is found in the approved configuration identification. The approved functional configuration identification is the functional baseline plus any approved changes from the baseline and serves throughout the system's acquisition life cycle as the current contractual description of the system's required functional characteristics.

5.2.2 Allocated Baseline.

Once the top-level system functional requirements have been generated, the contractor begins working on the development of the specifications for each of the CIs. (Often, the process of finalizing the System Specification and of identifying CIs and preparing initial draft specifications for those CIs will overlap.) An allocated baseline is established for each CI to more completely define and document the functional

characteristics of that configuration item as a part of the overall system or higher-level CI. The specification used to establish the allocated baseline defines the interfaces that must exist between the CI and other systems or CIs. It specifies the performance characteristics and design constraints that pertain to this specific CI as a part of the system, including the verifications which must be performed to show compliance to the performance characteristics.

For hardware CIs, the allocated baseline is normally established after the System Design Review (SDR) but no later than the Preliminary Design Review (PDR). For computer software CIs, the allocated baseline is normally established at the completion of the Software Specification Review (SSR), but it may be delayed until after the PDR for the CSCI. This baseline is established by the authentication of Type B [Development (for hardware CIs) and Software and Interface Requirements (for computer software CIs)] Specifications. The technical contract established with the allocated baseline provides the basis for the contractor to proceed into detailed design, development, building of test prototypes, and testing of the CI.

Computer software CIs establish their allocated baseline based on a software requirements analysis which defines and analyzes the functional, performance, interface, and qualification requirements for each CSCI. These requirements are derived from the functional baseline established when the System Specification or System/System Segment Specification was authenticated and contractually incorporated. Primarily, as far as software is concerned, the contractor is assuring that the characteristics of the computer system within the overall system requirements are described. The tasks associated with the Software Requirements Analysis are discussed in DOD-STD-2167. However, the documents which are produced that

establish the allocated baseline for the CSCI include the Software Requirements Specification and possibly the Interface Requirements Specification.

5.2.2.1 <u>Allocated Configuration Identification.</u> The allocated baseline plus any approved changes from this baseline constitute the approved allocated configuration identification (ACI). The ACI serves throughout the CI's life cycle as the current contractual description of the configuration item's required functional characteristics and is used to govern the development of the selected CIs.

5.2.3 <u>Product Baseline.</u>

After the contractor completes the design and testing of each CI/CSCI and successfully demonstrates that the CI/CSCI meets the previously stated technical requirements, the contractor finalizes the product specification and the associated detail design documentation (e.g., drawings, parts lists, source code) that will be used to establish the product baseline. However, the product baseline specifies more than just the physical design of the subsystem/CI; it also prescribes the required performance characteristics that should be tested during production and the acceptance tests to verify these characteristics. For computer software CIs, the product specification will include the updated Software Design Document and the listing of the source code from the end of formal qualification testing. The product baseline is normally established with the completion of the physical configuration audit; it normally includes Type C (Product) Specifications and may also include Types D (Process) and E (Material) Specifications.

5.2.3.1 <u>Product Configuration Identification.</u> The product baseline plus any approved changes from this baseline constitutes the approved product configuration identification

(PCI). The PCI serves throughout the CI's system acquisition life cycle as the current contractual description of its required detail design characteristics.

5.2.4 Developmental Configuration.

As shown in Figure 5 for each computer software CI (CSCI) after its allocated baseline has been established and prior to establishment of its product baseline, the contractor is required to internally control the evolving software design documentation during software development using the Developmental Configuration. The Developmental Configuration describes the CSCI's design and consists of documentation defining the code for the CSCI, its top-level and lower-level computer software components (CSCs), and its computer software units (CSUs). In addition, the Developmental Configuration also contains the complete and current software code (source and object) of all CSUs that have been successfully tested and reviewed. The Developmental Configuration is normally established at the completion of the computer software's preliminary design phase (with a successful Preliminary Design Review) and incorporates the Software Design Document(s) (Preliminary Design). The Developmental Configuration is continued through the CSCI's software developmental phases of: (1) detailed design, (2) coding and CSU testing, (3) CSC integration/ testing, and (4) CSCI testing phases. The activities required in these phases are described in depth in DOD-STD-2167. The following paragraphs provide a brief description of the processes on-going during these phases.

5.2.4.1 Preliminary Design. During this phase, the contractor is developing a preliminary design for each CSCI, allocating the CSCI's requirements from the Software Requirements Specifications and associated Interface Requirements

Specifications to the CSCs, and establishing the design requirements for each CSC. This information is documented in the CSCI's Software Design Document. In addition, if the CSCI must interface with one or more other hardware or computer software CIs, then the preliminary design information describing those interfaces is documented in a preliminary Interface Design Document. Finally, the contractor begins establishing the test requirements for conducting CSC integration and testing and identifying the qualification tests that will be required to show that each CSCI complies with the requirements listed in the Software Requirements Specification(s).

5.2.4.2 Detailed Design. During this phase, the contractor is developing a modular, detailed design for each CSCI and is allocating the requirements from each of the CSCI's computer software components to its computer software units (to include establishing the design requirements for these CSUs). In addition, if the CSCI externally interfaces with one or more other hardware or computer software CIs, then the detailed design of these interfaces is also developed. During this detailed design process, the contractor will be producing the design documentation plus preliminary test documentation (to include the test requirements, responsibilities, inputs, expected outputs, evaluation criteria, and schedules for CSC integration/testing, and CSU testing) and manuals. After successful completion of the Critical Design Review(s) for each CSCI, the contractor will incorporate the Software Design Document (Detailed Design), and, if necessary, an Interface Design Document for the CSCI into its Developmental Configuration.

5.2.4.3 Coding and CSU Testing. During this phase, the contractor will write executable code for each CSU, and then test the code to ensure that the algorithms

and logic employed are correct and allow the CSU to satisfy its requirements. These CSUs are the smallest logical entities which completely describe a single function in sufficient detail so that its code can be produced and tested independently of other CSUs. After each coded CSU is successfully tested and evaluated, its updated portion of the Software Design Document, and CSU source and object code listings are entered into the appropriate Developmental Configuration.

5.2.4.4 CSC Integration and Test. During this phase, the contractor will write the additional executable code (algorithms and logic) required to integrate the CSUs that comprise each CSC, such that the CSC, when executed, is able to satisfy its specified requirements. In addition, the contractor begins to integrate the CSCs and starts performing informal tests at the CSC level. The results of all CSU integration tests and all informal CSC integration tests for each CSCI are presented at a Test Readiness Review for the CSCI (see paragraph 6.1). For each successfully tested and evaluated CSC, the contractor will incorporate its updated portion of the Software Design Document, source and object code listings, and any other associated listing into the CSCI's Developmental Configuration.

5.2.4.5 CSCI Testing. During this phase, the contractor performs formal qualification tests on each CSCI to show that the CSCI will satisfy the specified requirements in its Software Requirements (and, if applicable, the Interface Requirements) Specifications. During the testing, if any problems occur, the contractor must make any necessary revisions to the Software Design Document and to the affected CSCI, CSC, or CSU source and object code listings. Upon successful completion of the formal qualification tests, the contractor must identify the exact version/revision of each CSCI and

document this information, along with the source and object codes, in a Version Description Document. With the results reported, a functional configuration audit is held for each CSCI to verify its performance. Following successful completion of the functional configuration audit, the Software Design Document and the source code listing for the CSCI are incorporated into the CSCI's Software Product Specification, which after successful completion of a physical configuration audit, is used to establish the CSCI's product baseline. When the product baseline is established, the CSCI's Developmental Configuration is no longer needed and will cease to exist.

5.2.5 Precedence.

To ensure that the specified requirements and the required functional, physical, and interface characteristics for a system are successfully allocated down to the configuration items, and that these configuration items are furthermore designed in the detail to allow them to meet their prescribed performance characteristics: the functional, allocated, and product configuration identifications should be compatible and consistent. If conflicts between the configuration identifications (baselines) go unresolved, the ability of the system to meet the overall stated requirements could be seriously hampered. The inability, for any lower subassembly, assembly, or configuration item, to concisely define its required functional or performance characteristics could lead to the failure of the system or system segment. Therefore, should any such conflicts occur between these configuration identifications, unless otherwise specified by the contracting agency through the contract, the order of precedence among the three is (a) functional, (b) allocated, and (c) product.

5.3 Configuration Identification Documentation.

The preceding paragraphs described the use of baseline management to contractually define detailed functional and physical descriptions of the hardware and software CIs. However, there has to be a vehicle used to establish these baselines and to provide a departure point for future development. The configuration identification of the system is the vehicle (documentation) that defines the "current baseline." To successfully employ baseline management, the program office and contractors must ensure that all of the needed functional and physical descriptions of the system or configuration item that will be used to prescribe the product's performance and design requirements are incorporated into the appropriate configuration identification documentation, namely, specifications, computer software listings, and drawings.

5.3.1 Specifications.

Specifications are the principal configuration identification documents prepared to support the acquisition of items. They describe the required functional and physical characteristics of the system and CIs in terms of design details, interfaces, performance, and system-related constraints (such as logistics or personnel constraints). Specifications also delineate all of the verifications required to demonstrate achievement of the functional, physical, and interface characteristics. The specifications are generated as a part of the system engineering (design) process. It begins with the analysis of the operational requirements listed in the System Operational Requirements Document (discussed in Section 3), or similar top-level requirements document, which is used to generate the top-level specification (usually a system specification) for the program. As further levels of functional analysis are

completed, the requirements are flowed down from the top-level specification to the specifications for the lower-level CIs (an example is shown in Figure 6).

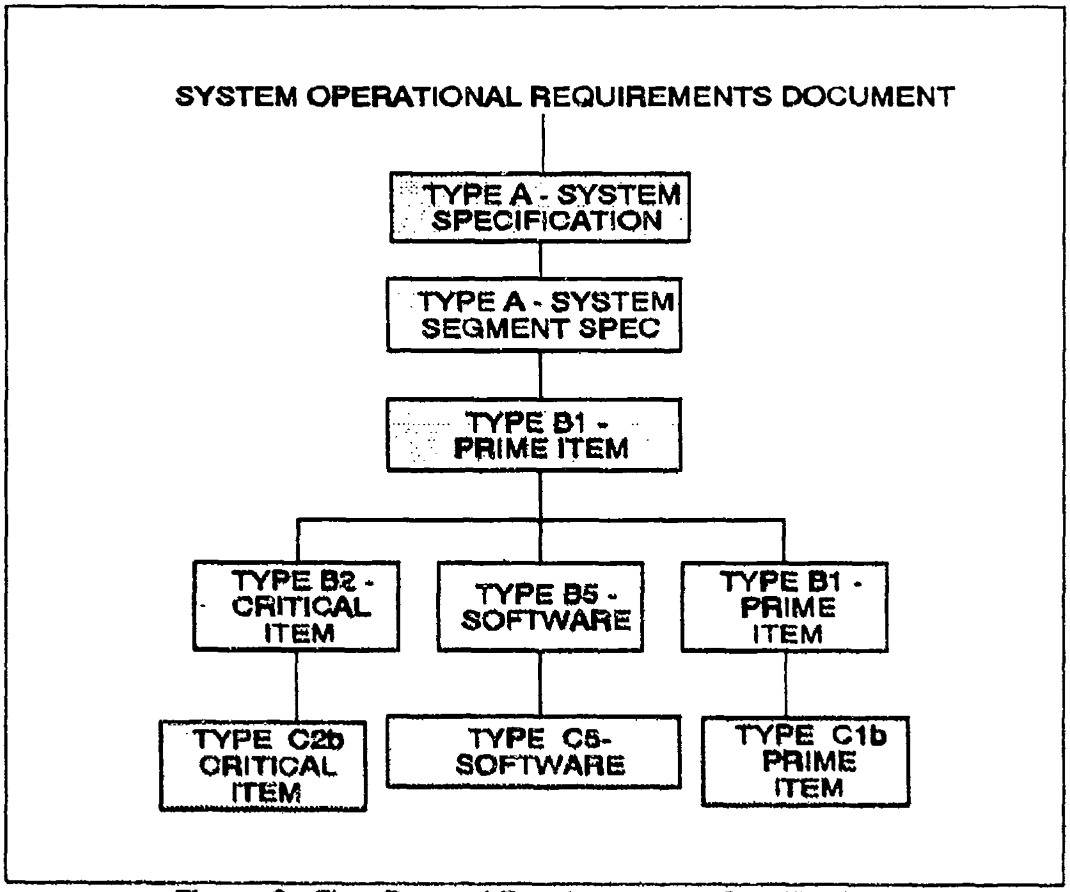

Figure 6: Flow Down of Requirements to Specifications

The following paragraphs discuss the various types of specifications that may be used as the product goes through the development process. [MIL-STD-490 prescribes the content and format requirements for the various types of specifications discussed.] These specifications are prepared to define the requirements that the item must meet. They are provided as a series of requirement statements which must be written in clear and simple contractually binding language. They should not contain general descriptive matter or vague, indefinite, redundant, or ambiguous requirements. In addition, specifications should not contain general contractual (tasking rather than

technical) requirements including management, procedural, or Statement of Work tasks (e.g., delivery quantities, schedules, costs, warranty provisions, or disposal instructions). The configuration manager is the focal point in the program office responsible for coordinating the review of the drafts to ensure that the specifications being developed meet these requirements. This is performed by making sure the specification(s) is reviewed to: (1) verify that all the required paragraphs are included; (2) evaluate the detailed technical content in each functional area, (3) verify the adequacy of testing specified, including a determination that the testing matrix is complete, (4) cross check the listing of the referenced documents in Section 2 of the specification, and (5) ensure format requirements of MIL-STD-490 are met. When the contents have been verified, the specification is then ready to be approved, authenticated, and baselined at the appropriate time.

5.3.1.1 Type A - System/System Segment Specifications. The System/System Segment Specification (SSS) specifies the functional, performance, and interface requirements for either a system or a segment of a system. The SSS states the technical and mission requirements for the system/system segment as an entity, allocates these requirements to functional areas, and defines the interfaces (internal and/or external) among the functional areas and with other systems. Additionally, the SSS specifies the requirements and any constraints for the system in areas such as computer resources and software; logistics; quality factors (e.g., reliability, maintainability); design and construction; and personnel and training that affect the total system.

The system specification is generated during the concept exploration/definition phase of the system acquisition life cycle. This specification, when baselined,

establishes the contractual performance requirements for the overall system. Once baselined, the system specification may be changed/revised via an Engineering Change Proposal (see paragraph 7.1.1.2) as the requirements and detail design are further defined, developed, and finalized over the remaining phases of its acquisition life cycle.

In addition to the system specification, a system segment specification may be required when the procurement of a very large system is apportioned among different program offices and/or different associate contractors. It may also be required when a system is acquired on an incremental or evolutionary basis, or where segments of an existing system are to undergo major modifications, or where a major segment is being developed to be added to an existing system.

The system, and/or system segment, specifications are used to establish the functional baseline for a program usually at the end of concept exploration/definition phase or early in the concept demonstration/validation phase. Once the functional baseline has been established, the SSS is maintained current throughout the development of the system as the overall basis for development and production of the system configuration items. The SSS is normally prepared by the contractor and authenticated by the Government, although the Government often prepares the preliminary draft system/system segment specification to accompany the Request For Proposal.

5.3.1.2 Type B - Development Specifications. These specifications are required as a result of the identification of various CIs and the need to allocate and expand design requirements from the system level to the CIs. They contain the performance requirements in the greater detail (than the requirements in the SSS) needed to define

the expected CI performance, but they should minimize the amount of specific design information. In addition to these required performance requirements, the development specification may also address the external and internal interfaces for the CI, required standard (Government furnished equipment) components, logistics support, personnel and training requirements, and qualification testing desired for the CI. Each development specification is used to establish each CI's allocated baseline, which is then maintained as a complete statement of all contractual requirements. The breakdown of a system into its subelements will involve CIs of different degrees of complexity. More complex CIs are more likely to be subjected to requirements relating to more different engineering disciplines; so the specification content required by MIL-STD-490 is greater for the more complex CIs. To accommodate the variety of CI complexities, the development specifications are classified into subtypes. Some of the characteristics for these sub-types are given in the following paragraphs.

5.3.1.2.1 Type B1 - Prime Item. This sub-type is also referred to as a prime item development specification (PIDS). The PIDS is applicable to the highest level, most complex items in the system, such as an aircraft, missile, or launcher equipment. Basically, the PIDS provides a delegation (allocation) of the system-level operational requirements to the prime item. The PIDS establishes the requirements and constraints for performance (manning, operating, maintenance, and logistics support), design, principal interfaces (functional and physical), and verification requirements. The functional interfaces are usually specified in quantitative terms while the physical interfaces are usually expressed in terms of dimensions with tolerances. It also identifies, and discusses the use of, Government-furnished property to be designed into, and delivered with, the prime item or to be used with the prime item.

The Type B1 Specification may be used as the functional baseline for a development program for a single, small CI or as part of the allocated baseline where the CI covered is part of a larger system development program. The CIs that require a Type B1 Specification meet the following criteria:

1. The prime item is received or formally accepted by the Government with a DD Form 250.

2. The item requires quality conformance inspection and/or acceptance tests at the prime item level of assembly.

5.3.1.2.2 Type B2 - Critical Item. This sub-type is referred to as a critical item development specification (CIDS). The CIDS establishes the performance requirements, design constraints, and verification requirements for a subsystem/ assembly which is below the level of complexity of a prime item but which, for technical management reasons (engineering critical) or supportability reasons (logistics critical), is selected as requiring separate identification and control. An item is considered engineering critical if: (a) the technical complexity/risk warrants the expansion of the item requirements defined in the higher-level CI specification into its own specification, (b) the reliability of the item will significantly affect the ability of the system to perform its overall mission, or (c) the successful performance of the higher-level prime item cannot be adequately evaluated without some separate evaluation and testing of the proposed critical item. To be considered logistics critical the item must: (a) require significant numbers of different repair parts to be purchased to support it, or (b) be designated for multiple source reprocurement and require the separate documentation in order to support such reprocurement.

5.3.1.2.3 Type B3 - Non-Complex. This sub-type is used for non-complex items, such as special tools and work stands, which require documentation of constraints for development and qualification purposes but which, once development is complete, need only simple demonstrations and/or comparison with drawings prior to delivery. As such, the production units of the item should not require acceptance testing to verify their performance. Not all non-complex items need specifications; many can be adequately developed and manufactured using appropriate engineering drawings.

5.3.1.2.4 Type B4 - Facility/Ship. This sub-type is called a facility or ship specification and is used for an item's development only. The item involved would be a facility or carrier vehicle (e.g., a ship, airplane, tank) which is an integral part of the overall system. Examples include the Pave Paws submarine-launched missile detection system facilities and the Peacekeeper missile system launch control facilities. This specification establishes the requirements and basic constraints, plus the verifications, imposed on the development of an architectural and engineering design for an item that allows it to integrate the capabilities of the major system. The critical interfaces that must exist between the facility and other system functional elements are also defined in this specification.

5.3.1.2.5 Type B5 - Software. These following two subtypes focus on the requirements for the computer software that is being designed as a part of the system or higher-level hardware item.

1. Type B5a - Software Requirements Specification (SRS): The SRS is used to provide the detailed requirements for each CSCI. The requirements allocated to the CSCI and expanded in the SRS will provide the basis for the Government to assess

whether or not the completed CSCI design works correctly and effectively. The contractor uses the SRS as the basis for the development and formal verification of the CSCI.

2. Type B5b - Interface Requirements Specification (IRS): The IRS specifies, in detail, the requirements for one or more external interfaces that must exist between a particular CSCI and other hardware and computer software or other systems. These specified requirements alert the contractor to focus on the design, development, test, and evaluation of certain interfaces for the required CSCI. However, Interface Requirements Specifications are not always required. For most systems involving software, interface requirements should be included in the SRS, especially in situations where (a) there are few interfaces for the CSCI, (b) the interfaces are simple, or (c) there is only one contractor developing the software. However, in situations where (a) there are numerous external interfaces for the CSCI, (b) the interfaces are complex in nature, or (c) there is more than one contractor responsible for the development of the system software, an IRS may be required. The contractor uses the IRS as a basis for the development of these interface(s). The Government uses it to assess whether or not the required interfaces have been achieved.

5.3.1.3 Type C - Product Specifications. These specifications are also used (like the Type B Specifications) for CIs below the system level. They are usually companions to the Type B Specification for the CI/CSCI. Most Type C Specifications are oriented toward the procurement of a product and normally define its physical, fabrication, and detailed design requirements. The product fabrication specification prescribes the detailed description of the parts and assemblies of the CI, using its drawings and parts list, and the acceptance tests and inspections required to assure proper fabrication and

assembly techniques. For software CIs, the product specifications include design document(s) and the corresponding source code to define the detailed design. Some Type C Specifications are oriented toward just prescribing functional requirements; in that case, the specification covers the form, fit, and function of the product (external dimensions and detailed performance requirements) for its intended use, qualification and acceptance tests, and, where appropriate (prime items only) its interface and interchangeability characteristics. The following paragraphs discuss the various sub-types of the product specification.

5.3.1.3.1 Type C1 - Prime Item. These are the prime item product specifications applicable to those CIs that meet the criteria for prime items as discussed in paragraph 5.3.1.2.1. The Type C1 Specifications may be prepared either as a product function (performance-oriented) or fabrication (detailed design) specification depending upon the procurement conditions.

1. The Type C1a, prime item product function specification, establishes the performance, manufacture, interface (functional and physical), and interchangeability requirements and the qualification and acceptance provisions for the prime item. This sub-type specification is used when definition and control of the specific internal design of the item is not critical, and when training and logistics considerations are considered unimportant. (Normally, the item will be thrown away or repaired by other than Department of Defense personnel.)

2. The Type C1b, prime item product fabrication specification, establishes the requirements for the manufacture and acceptance of prime items whose basic performance requirements are defined in a related Prime Item Development Specification. This sub-type of specification is prepared for the procurement of prime

items when: (a) control of a detailed design disclosure package is needed, (b) the interchangeability of lower level components and parts needs to be controlled, and (c) maintenance by, and training of, DOD personnel is expected to be a part of the prime item's operational employment. It must also contain and/or reference all drawings and documentation required to completely describe the specific design.

5.3.1.3.2 Type C2 - Critical Item. This sub-type includes the critical item product specifications that are applicable to those engineering or logistic critical CIs that meet the criteria of critical items discussed in paragraph 5.3.1.2.2. The Type C2 Specifications may be prepared either as a product function (performance-oriented) or fabrication (detailed design) specification depending upon the procurement conditions.

1. The Type C2a, critical item product function specification, establishes the performance and manufacture requirements and the qualification/acceptance provisions for the critical item. It provides only a form, fit, and function description of the CI. It is used when the performance characteristics are of more concern than part interchangeability or control over the details of the design.

2. The Type C2b, critical item product fabrication specification, establishes the requirements for the manufacture and acceptance of a critical item whose basic performance requirements are defined in a related CIDS. This type of specification is used when a detailed design disclosure needs to be available or where it is considered that adequate performance can be achieved only by adhering to detailed drawings and processes.

5.3.1.3.3 Type C3 - Non-Complex Item Product Fabrication. This sub-type applies to those CIs which meet the criteria for non-complex items discussed in paragraph

5.3.1.2.3. It establishes the requirements for the manufacture and Government acceptance of non-complex items whose performance requirements are defined in a related non-complex item development specification. Normally, it is used to define the detail design of a product that will be accepted simply by inspecting it against the drawings.

5.3.1.3.4 Type C4 - Inventory Item. This sub-type is used to specify the requirements for inventory items that are available in the Government inventory (e.g., tools, computers, computer programs, and installed equipment) and are to be incorporated in the computer software CIs and prime item CIs of the system. The inventory item specification is used to stabilize the configuration of inventory items on the basis of current capabilities of the item and the requirements for the specific application. This specification includes: (a) specific identification of each of the key pieces of Government-furnished equipment, including a reference to its military specification, (b) the general requirements for all of these items, and (c) a separate appendix for each item that lists its critical performance and interface requirements. When appropriate, it will also specify any additional (or relaxed) requirements that may be related to the use of the item for this application. Usually, a single inventory item specification is sufficient. However, when large numbers of items are involved or when associate contractors are involved, a separate inventory item specification may be prepared, as required, for each system segment or prime item in which inventory items are to be installed.

5.3.1.3.5 Type C5 - Software. This is the Software Product Specification (SPS) and is applicable to the delivered, as built, CSCI. The SPS includes the design documents

and source code listings for a CSCI. Until the product baseline is established, these contents of the SPS are contained in the contractor's Developmental Configuration for the CSCI. Upon successful completion of the CSCI's physical configuration audit, the product baseline is established and the SPS is finalized to include the final up-dated versions of the following documents:

1. Software Design Document (SDD): The SDD describes the structure and organization of a particular CSCI. It documents the allocation of the CSCI requirements specified in the Software Requirements Specification and, if applicable, the Interface Requirements Specification(s) down to the computer software components and from these CSCs to their associate computer software units. The SDD also defines the interface, data, and processing characteristics for the CSCI to its CSCs and to the CSUs. Any engineering information (design rationale, results of analyses, trade-off studies, or any other information deemed appropriate) that was generated during the preliminary and detail design process of the CSCI may also be incorporated into the SDD.

2. Interface Design Document (IDD): When applicable (when one or more CSCI interfaces exist in the system and require an IRS), the IDD describes the detailed design of the external interfaces between the CSCI and other CSCIs and hardware CIs. The IDD documents the details of the information that is passed via the interfaces and reflects the results of the interface design decisions.

5.3.1.4 Type D - Process Specifications. This specification is used to define the details of a peculiar treatment or process (e.g., one developed specifically for this system like a heat treatment) that is critical to the manufacture of the system. Normally, the contractor's processes are controlled through their citation on the

controlled drawings, and a Type D Specification is not required. If needed, the Type D Specification usually defines the complete step-by-step procedure used by the manufacturer in order to assure that a satisfactory result is achieved. Normally, the process specification is incorporated into the product baseline, but it may also be prepared to control the development of a process as a part of the allocated baseline.

5.3.1.5 Type E - Material Specifications. This specification is used to define the details of the raw materials (e.g., chemical compounds) and mixtures (e.g., propellants, paints, cleaning agents) that are critical to the fabrication of the system. It identifies the actual minimum functional, physical, chemical, electrical, and/or mechanical requirements of the material. The requirements are specified in sufficient detail to allow reproduction of the material without having to use or consult the original manufacturer. Normally, the material specification is incorporated into the product baseline, but it may also be prepared to control the development of the materials and mixtures used in the manufacture of the item as a part of the allocated baseline.

5.3.1.6 Two-Part Specifications. For most CIs, both a development and a product specification will be prepared and authenticated. While they may be controlled and maintained as separate specifications, the program office may opt to combine both as a single specification document. Under this practice, the development portion of the combined specification is referred to as Part I while the product fabrication portion is referred to as Part II. Under this concept, Part I of the specification is written, reviewed, approved, and authenticated as a separate specification. The Part II portion of the specification is initially written and reviewed in draft form as a separate document, but it will have the same specification number as the Part I portion. When

this Part II portion is finally approved and authenticated, it is combined with the Part I portion, and they are bound together and controlled as a single specification. Using this practice, the performance requirements in Part I are maintained throughout the life cycle as current contractual requirements that must be achieved by the product design specified in Part II. Thus, any proposed design changes must be evaluated against both the product fabrication and the development parts of the specification. The use of a two-part specification is not applicable when using a product function specification, but they may be used with computer software specifications.

5.3.1.7 Forms of Specifications. Program specifications will be prepared in accordance with the requirements of MIL-STD-490 to the degree required by one of the following forms, as specified in the Contract Data Requirements List in the contract.

5.3.1.7.1 Form 1. This form provides Specifications to Military Standards. Under this concept, the program specifications are required either to conform exactly to MIL-STD-490 (called Form 1a) in all details, including paragraph numbers and paragraph titles, or they must conform to most of the requirements in MIL-STD-490 (called Form 1b) with regards to section numbers and titles (specification content), but not with specific paragraph sequencing, numbering, or titling. Most program specifications, particularly those which may be used for future competitive reprocurement, are ordered Form 1a or Form 1b.

5.3.1.7.2 Form 2. This form provides Specifications to Commercial Practices with Supplemental Military Requirements. This form requires the contractor to comply with technical society standards, normal industry standards, or MIL-STD-490. In addition, specifications of this form should meet, at a minimum, the following requirements:

1. It should specify sufficient requirements to assure ease of procurement and delivery of like materials, products, and services using the specification.

2. It should specify sufficient qualification and acceptance test requirements to assure conformity of the item to the specified requirements.

3. It should include a cross-reference between Government standards and industry or contractor standards when the latter are cited in the specification. [At this writing, MIL-STD-490 is in the final stages of revision; as it incorporates the Form requirements, it is likely that only Form 1a, Form 1b, and Form 3 (redesignated Form 2) will remain.]

5.3.1.7.3 Form 3. This form provides Specifications to Commercial Practices. This form allows the contractors to prepare the specifications in their own format using their normal practices.

5.3.2 Drawings.

Another major category of the technical documentation that constitutes a part of an item's configuration identification is engineering drawings. The types of drawings required for a CI's complete identification, and the amount of detail information they must contain, depends upon the complexity and sophistication of the design. For older programs which were developed using military specification DOD-D-1000 (Drawings, Engineering, and Associated Lists), there are three Levels of drawings that may be used for a part/item, depending upon what function the drawing will be asked to perform in the future. These Level designators for the drawings allows for, and coincides with, the progression of the design through the demonstration/validation, full-scale development, and production phases of its acquisition life cycle. For newer

programs, specification MIL-T-31000 (Technical Data Packages, General Specifications For) provides for three basic drawing options (conceptual design, developmental design, and product) which are very similar to the three Levels of drawings.

5.3.2.1 Level 1, Conceptual and Developmental Design [DOD-D-1000]. Drawings at this Level provide basic design information in either the conceptual design or the developmental design for an item. To reduce technical uncertainty in the conceptual design of an item, this Level of drawing allows for the evaluation of the feasibility of an engineering concept or technology and for the evaluation of the utility of the design in meeting the operational requirements. For a developmental design, this Level of drawing provides the information necessary to fabricate developmental hardware and/ or prototype components. In addition, the data associated with these developmental drawings should be adequate to allow an analytical evaluation of the inherent ability of the design to meet the required performance. The format for these drawings may vary from informal sketches to formally prepared drawings.

5.3.2.2 Conceptual Design Drawings and Associated Lists [MIL-T-31000]. These drawings describe the engineering concepts on which a proposed technology or design approach is based, and define the design concepts in graphic form (including any appropriate textual information required for analysis and evaluation of the concept). They are used when there is a need to verify the preliminary design and engineering and to confirm that the technology being suggested is feasible and that the design concept has the potential to be useful in meeting the stated operational requirement.

5.3.2.3 Level 2, Production Prototype and Limited Production [DOD-D-1000]. These drawings contain more detailed design information than those in Level 1 and must be

prepared in accordance with the requirements of DOD-STD-100. Drawings that are prepared to this level disclose a design approach suitable to support the evaluation of the proposed production design. They contain enough detailed manufacturing information to support the building of prototype items for testing and may be used to support low-rate initial production of the items. The items built to this level should be suitable for field test, deployment, and logistics support. Although these drawings may lack some of the detailed information that would allow for competitive reprocurement of an item, any necessary special inspection and test requirements that would be needed to verify the items compliance with its requirements must be defined on the drawings or referenced to a document that is available to the Government.

5.3.2.4 Developmental Design Drawings and Associated Lists [MIL-T-31000]. These drawings describe the physical and functional characteristics of a specific design approach and provide sufficient data to the extent necessary to support the analytical evaluation of the specific design approach to meet the specified requirements. In addition, they should enable the development and fabrication of any prototype hardware for test or experimentation, if deemed necessary by the contractor or program office. These drawings may vary from simple sketches to complex drawings, or any combination in-between.

5.3.2.5 Level 3, Production [DOD-D-1000]. Drawings prepared at this Level are formally prepared to DOD-STD-100, and all processes and standards referenced by the drawing must be either military or industry standards or must be provided with the drawing. (Essentially, Level 3 is a DRAWING PACKAGE rather than just a drawing.) These types of drawings provide the highest degree of definition of the design to the

Government. This Level of drawings (along with the documentation in the Level III package) provides sufficient engineering definition to enable a competent manufacturer to produce the item detailed, without recourse to the original designer, in such a manner that its physical and performance characteristics are identical to those of the originally designed and produced items. This desired capability requires that the drawings include details of unique processes; dimensional and tolerance data; critical manufacturing assembly sequencing; mechanical and electrical connections; physical characteristics (form and finish); calibration information; and inspection, quality control, and test criteria. Inclusion of such information in the Level 3 drawings provides control of the end product design, provides detailed engineering data for the support of a quantity production run, and permits competitive reprocurement of the item.

5.3.2.6 Production Drawings and Associated Lists [MIL-T-31000]. These drawings provide the necessary design, engineering, manufacturing, and quality control information necessary to permit a competent manufacturer to produce an interchangeable item which duplicates the physical and performance characteristics of the original design without additional design engineering or recourse to the original manufacturer. They reflect the maturity that the design of the item has attained and are selected whenever there is a current or future need for the Government to reprocure or manufacture the equipment, components, or spares and repair parts.

5.3.2.7 Types of Drawings. Engineering drawings are documents that disclose, through pictures and text or a combination of both, the physical and functional requirements of the end item. There are a number of different types of engineering drawings that must be provided by the contractor in order to provide the complete

detail information about the design. DOD-STD-100, Engineering Drawing Practices, discusses the 40-odd different types of drawings which may be ordered as Level 2 or Level 3 and provides the details for the content of these drawings. Among the types of drawings discussed in DOD-STD-100 are the following.

5.3.2.7.1 Detail Drawings. These drawings, which may be either mono- (describing a single part) or multi- (describing more than one part) detail, depict the exact design for the part delineated on the drawing and are suitable for use in fabricating the part. Detail drawings depict, at the minimum, the part(s)'s dimensions, tolerances, materials, mandatory manufacturing processes, surface finish, and any protective coating. Detail drawings are required by a second source contractor in order to actually manufacture the part or item.

5.3.2.7.2 Assembly Drawings. These depict the assembled relationships of (a) two or more parts, (b) a combination or parts and subordinate assemblies, or (c) a group of assemblies required to form an assembly of some higher order. The drawings should include several views, if needed, to show the assembly relationships between all subordinate assemblies and parts which comprise the assembly being depicted. These drawings cannot be used by themselves for reprocurement, nor can they be used for manufacturing the individual parts that make up the assembly being shown. They are needed in a Level 3 package, in concert with the detail drawings, to show how the individual parts shown on the detail drawings fit together. Through parts lists and part numbers (by part or item), they provide a guide to the part's detail drawings.

5.3.2.7.3 Control Drawings. These are engineering drawings that disclose requirements such as external configuration, expected performance, test requirements,

and weight and space limitations to the extent that an item can be procured in the commercial market to meet the stated requirements. There are different types of control drawings, some of which are:

1. Specification Control Drawings: depict existing commercial or vendor-developed items which are advertised or catalogued as available on an unrestricted basis. The drawing discloses physical (i.e., configuration, dimensions) and functional (i.e., performance, inspection, and test) characteristics to insure adequate requirements are available for use in the reprocurement of interchangeable items. It is not a detail drawing and does not provide sufficient information to allow for the direct fabrication of the item. These drawings should not be prepared to so uniquely identify the characteristics such that the Government is restricted to the procurement of a single vendor's item to the exclusion of other equally suitable products. The drawing contains a nonrestrictive (not intended to represent the only sources) list of suggested sources, each of which would have their own detail drawing (with different identifying numbers) for manufacturing the item.

2. Source Control Drawings: depict an existing commercial or vendor item which provides the performance, installation, and interchangeable characteristics required for specific critical applications. It contains data/requirements/tests similar to those of the specification control drawings. The major difference between these type of drawings is the source listing. For the source control drawing, the sources listed are the <u>only approved sources</u> for the part depicted. This is because the parts listed have been specially tested and approved for use in the specific applications that are stated on the drawing. Other similar, but untested, items may cause the system to fail prematurely.

Whenever another vendor's item is qualified for the stated application(s), it too may be added to the source list.

5.3.2.7.4 Installation Drawings. These should show the general configuration and complete information necessary to install an item relative to some support structure or other item. The drawing should include any interface information, pipe and cable attachment information, and the principal dimensions that establish the limits in all directions for the item's installation.

5.3.2.7.5 Diagrammatic Drawings. These delineate features and relationships of items forming an assembly or system by means of symbols and lines. They are graphic explanations of the manner by which an installation, assembly, or system performs its intended function. Some of the different types of diagrammatic drawings are:

1. Schematic Diagram: this shows the electrical connections and functions of a specific circuit arrangement. By using standard symbology, it allows an electronics specialist to understand and analyze the design by tracing the circuit and its functions without regard to the actual physical size, shape, or location of component parts.

2. Connection or Wiring Diagram: shows the electrical connections of an installation or of its component devices or parts. It may cover internal or external connections, or both, and contain the detail to make or trace connections involved. It shows general physical arrangement of the component devices or parts.

3. Interconnection Diagram: connection or wiring diagram which shows external connections between units, sets, groups, and systems.

4. Logic Diagram: uses graphics to show the sequence and function of logic circuitry (firmware). It is used to document the hardware components which relate to the computer program inherent in the design.

5.3.2.7.6 <u>Special Purposes Drawings.</u> These are other than end-product drawings that are used to supplement the requirements. These type of drawings may be required for management control, for logistic purposes, as manufacturing aids, or for configuration management of the item. Some special purpose drawings are:

1. Wiring List: this is an engineering drawing (usually in "Book- form") consisting of tabular data and instructions to establish wiring connections within or between units of equipment, sets, or assemblies of a system. It is a form of interconnection or connection diagram. A wiring list shows the starting and ending connections for wires interconnecting circuit boards and/or black boxes.

2. Wiring Harness Drawing: shows the paths of groups of wires connected together in a specified configuration to simplify installation. These drawings show the dimensions necessary to define the harness form and the termination points and include wire data tabulations, circuit designations, wire lengths, and material specifications. The drawing will also include instructions for the preparation of the harness and its associated schematic and wiring diagrams.

3. Book-form Drawing (So called because it is usually a multi-page, 8 1/2 x 11 inch document.): this is an assemblage of related data that discloses the engineering requirements for an item, a family of items, or a system. It uses either pictorial delineations, text, technical tabulations, or a combination of all three. Examples of Book-form drawings include wiring lists and parts lists.

5.3.2.7.7 Layout Drawings. These provide graphic depiction of design development but only to the extent that it shows a design solution. It does not establish the item identification.

5.4 Configuration Item Numbering.

Now that the different types of technical documents and drawings that constitute the configuration identification for each of the different baselines have been discussed, how do we manage and control these documents? The configuration management process requires that each CI (hardware and computer software) be uniquely identified at all times. One of the keys to successful management of CIs is identification numbering, both of documents (specifications and drawings) and CIs (hardware and computer software). This allows discrimination between documents/units with different contents and/or configurations. The use of a unique identifier for each functionally or physically different configuration of an item allows the identification to be absolute. In addition, uniqueness of identification numbers allows the program office to control, and maintain traceability of, each baselined CI and its approved configuration identification documents during the life cycle of the CI. The identification numbers are assigned and controlled by various agencies, including the contractor, in accordance with MIL-STD-482 and MIL-STD-483. Various types of identification numbers are issued for items and their documentation depending on the item requirements for engineering/configuration control and logistics/inventory control.

5.4.1 Engineering and Configuration Control.

5.4.1.1 Configuration Item Identification (CII). The CII number is a seven alpha-numeric character identifier which must be unique for each CI in a specific system.

The contractor selects, assigns, and issues the CII numbers without any formal approval from the procuring activity. However, the number must meet any requirements and restrictions listed in MIL-STD-482 and MIL-STD-483. The CII is printed on the cover sheet of specifications to identify the documentation as relating to that CI, and it is used on ECPs, deviations, and waivers to provide traceability to the CI affected. In addition, the use of a CII number:

1. Establishes a base for serializing individual units of a CI.
2. Does not change when the unit is modified (unless it is a major functional change).
3. Should remain the same for the CI even though there may be more than one application or more than one contractor involved.

5.4.1.2 Contractor and Government Entity (CAGE) Code. This may also be referred to as the Manufacturer's Code. This five digit alpha-numeric code is issued and controlled by the Government (using Cataloging Handbook H4/H8) and uniquely identifies federal suppliers doing business with the Government. This code is required on drawings, documentation, and all nameplates affixed to units delivered by the contractor. This allows the Government to determine who supplied the particular unit or document.

5.4.1.3 Part Number. This consists of up to fifteen alpha-numeric characters, established by the contractor, and must be unique for each part, assembly, and item designed at a particular (CAGE) contractor location. The design activity must not use the same part numbers for this or any other program. However, because there are some common methods of constructing part numbers, the same part number may be used by another (CAGE) design activity. The part number provides a unique identifier

through combination with the CAGE, which then provides a unique identifier for all parts in the Government inventory.

5.4.1.4 Serial Number. This type of identification number may be issued by either the Government or the contractor, depending on the type of item involved. Most serial numbers are issued by the contractor and contain at most fifteen alpha-numeric characters, with at least the last four being numeric. It is used to uniquely identify each delivered unit of important assemblies. The serial numbers are assigned in numeric sequence within the CI group and are permanent for the life of the unit. It must not be repeated for the CI/part number identified, but the same serial number may be used on other units of other CIs/parts with different configurations. For some critical assemblies of parts and specially-designated expendable items, the serial numbers may be issued by the Government to assure uniformity among the units being used in the inventory.

5.4.2 Logistics and Inventory Control.

5.4.2.1 National Stock Number. The National Stock Number (NSN) identifier is issued and controlled by the Cataloging and Standardization Center (Battle Creek, Michigan) and is used to facilitate the management of similar parts in the Government inventory. Because there are so many different ways that contractors may issue part numbers, it is possible to have two parts with exactly the same design/function but with massive differences in part numbers. Any "normal" sequential ordering of the part numbers would fail to identify these as interchangeable parts. Through the use of a NSN, such interchangeable parts are cataloged under a single National Stock Number. The NSN is thirteen numeric characters arranged in four fields; two of the fields (the first six

characters) have controlled (meaningful) content. Although there are no duplicate NSNs (for different designs) in the Government inventory, several vendor parts with interchangeable designs may have the same NSN.

5.4.2.2 Nomenclature Designation. A nomenclature consists of an identification number followed by a short word phrase describing the item so identified. A nomenclature is issued and controlled by the Government; it provides a unique, basically unchanging identifier for major assemblies of parts. It is as important for effective logistics and supply support of those assemblies as the NSN is for parts and small assemblies; and it provides a means of inventory control for common items. Nomenclatures are issued for an item upon receipt of a formal request from the design activity. The Air Force issues nomenclatures for airborne units (e.g., missiles, helicopters, aircraft); the Army issues nomenclatures for electronic equipment (e.g., radios, test equipment, computers); and the Navy issues nomenclatures for munitions items (e.g., rounds of ammunition, bombs, missile warheads). The nomenclature designation is required for all CIs and may also be needed for component sub-assemblies that are: (a) expected to be used in more than one system, (b) complex in design, (c) removable without desoldering, (d) repairable, or (e) able to change the performance of the basic unit.

The nomenclature designation is similar to an item's CII number. However, while all CIs must have a nomenclature, not all items with nomenclatures are CIs. The program office must make sure that the contractor submits requests for the nomenclatures for the CIs as soon as they are selected, since the nomenclatures are required to be included on the CI specification cover sheet. Nomenclatures should be requested for the overall system and its major CIs during concept demonstration/

validation and for the lower-level CI and their major assemblies during full-scale development.

5.4.2.3 Computer Program Identification Number (CPIN). CPINs are Air Force designators that are issued by the Oklahoma City Air Logistics Center (OC-ALC). A different CPIN is issued for each basic computer software program and for each subsequent revision or version in the operational inventory (including support equipment computer programs). The CPIN has a variable field length; it is segmented into sections that identify the host component/system, the type of computer software involved, its use, and designators reflecting the current version of the software. A unique CPIN will be used to identify each computer program; all documents (manuals, specifications, computer software listings) related to that particular computer software program will carry the same CPIN except for the change in one character to indicate that it is documentation. Thus, the Air Force is able to identify and interrelate its inventory of computer software and related documentation. Because the leading field of the CPIN designates the use/purpose of the computer software, the Air Force is also able to facilitate the reuse of software. Like items of computer software are grouped together (as with hardware NSNs), and a design activity developing a new system can review information about available programs, components, and units in monthly compendiums of Air Force computer software published by OC-ALC.

6. DESIGN REVIEWS AND CONFIGURATION AUDITS

As the acquisition program progresses through its life cycle, the design of the system becomes more and more defined. The systems engineering process helps identify the major subsystems, system segments, and CIs for the system and flows down the stated operational requirements to each CI, so that the CI can be designed in such a way as to fulfill its part of the system requirements. At appropriate times in the development process, the requirements for each CI are baselined and its configuration identification placed under control. However, the program manager must monitor the progress of the development of the requirements and of the detailed design to assess the ability of the CI design to achieve the stated requirements. This is accomplished by conducting technical design reviews and configuration audits as system and CI designs evolve. The reviews and audits are the means of monitoring the progress and readjusting the development of the system design if required. This section will outline and describe the different technical design reviews that may be held for a system, subsystem, or CI. Afterwards, this section will discuss the different types of configuration audits that the program office may employ to verify and validate the design.

6.1 Design Reviews.

There are a number of design (technical) reviews which are an essential part of the systems engineering process. Held after the contractor (design activity) has completed a specific increment of the design work, they provide the program office (management activity) with the visibility into, and a technical understanding of, the adequacy of the development effort to date in meeting the specified technical requirements. As the

acquisition program progresses through its life cycle, the design documentation becomes more detailed and the design reviews must accomplish increasingly detailed evaluations. Each design review has a specific purpose, but each must consider all aspects of the system, including all related specialty engineering disciplines, in evaluating the adequacy of the design effort/results at that point in the system's development. During the early design reviews, the configuration manager participates to assure that the draft documentation (specifications) being presented completely defines the technical requirements for the system and the CIs before that documentation is formally baselined. Later in the system/CI design and development, the configuration manager is primarily concerned with identifying design problems which may lead to engineering changes to the established baseline documents. The configuration manager also uses the design reviews as the means to assess the contractor's adherence to approved configuration management procedures at each point in the system acquisition life cycle.

The program office normally uses a series of formal reviews conducted in accordance with MIL-STD-1521 [NOTE: At the time of this writing, the design review requirements are to be incorporated into MIL-STD-499.]. The number of design reviews will be depend primarily upon the number of CIs selected; normally, there is a separate session of each type of design review for each CI. However, the amount of detail to be covered at each CI's session of the design review (and, to a certain extent, the number of separate reviews) is dependent on the technical complexity and technical risk associated with each CI. For example, if a CI is of a complex technical design, using a new technology in a critical functional area of the system, then the program office will normally hold a design review that focuses on that configuration

item only. On the other hand, if several lower-level CIs have a less complex design and lower technical risk, then their design reviews may be combined into one design review for the higher-level CI. The reviews will consider the design aspects, technical performance estimates, and engineering integration efforts for each (all) configuration item(s). In order to completely evaluate the progress, there must be participants from all affected engineering (technical) disciplines and representatives from all participating functional activities.

The formal reviews are not the only means of evaluating the contractor's progress. Often the program office will use informal reviews that may be called to resolve specific technical issues. Participation at these informal technical meetings (commonly referred to as Technical Interchange Meetings) is often restricted to program office personnel (and possibly supporting agency personnel) with specific expertise in the problem area.

Figure 7 illustrates the relationship of these formal design (technical) reviews (including the configuration audits) to the system acquisition life cycle and to baseline management concepts. (It is based on Air Force policy established by AFR 14-1.) Since the systems engineering process is iterative (paragraph 3.3), and since the development process varies from program to program, the design reviews cannot be tied to specific points in the system acquisition life cycle. However, the indicated points for conducting the design reviews and for establishing the corresponding configuration identification baselines shown in Figure 7 indicate the <u>normal expected points</u> in the life cycle when design reviews and baselines should occur for a given system, system segment, and/or CI. For example, the System Design Review (SDR, discussed in paragraph 6.1.2) and the establishment of the corresponding allocated

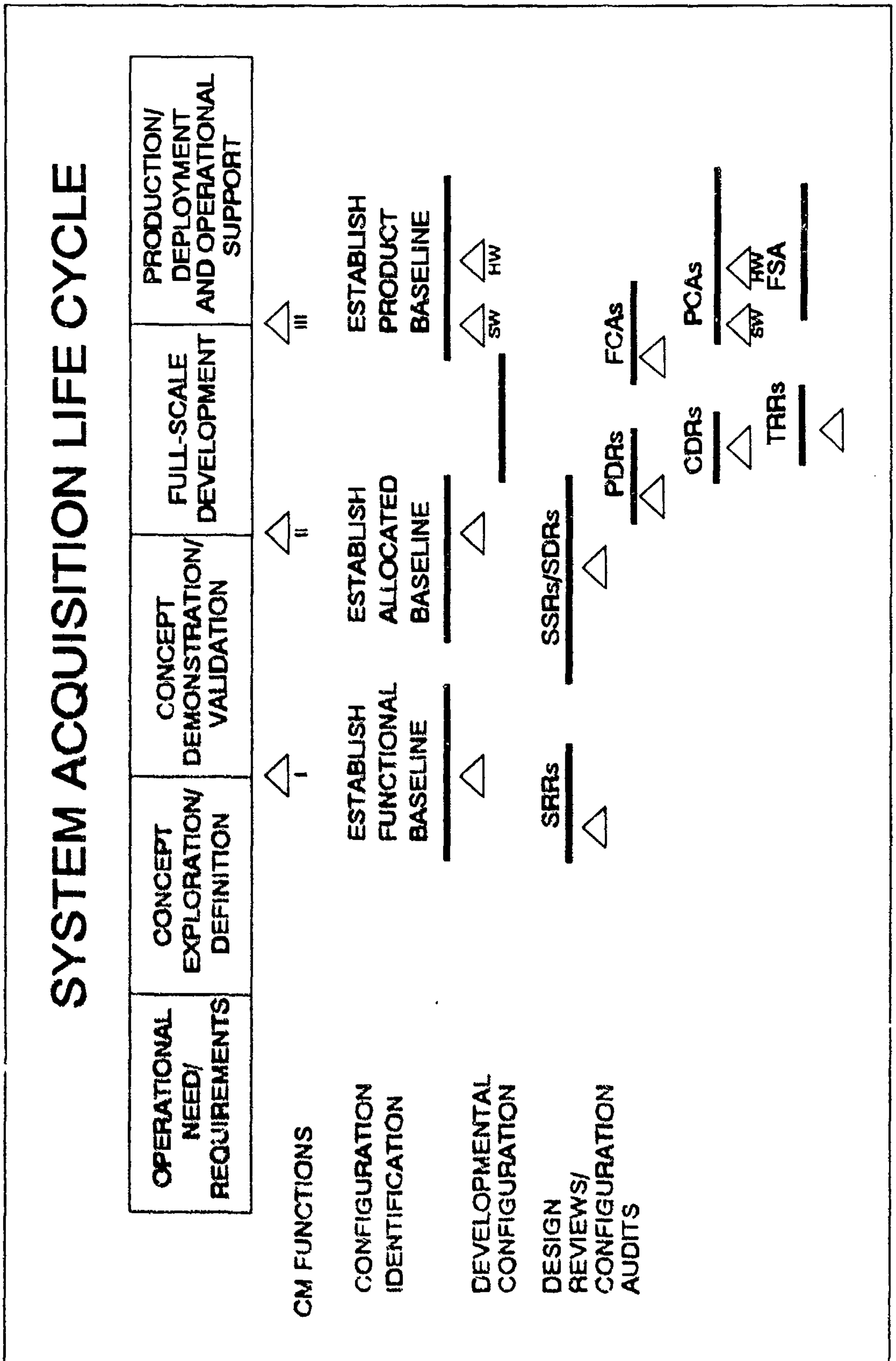

Figure 7: Design Reviews and Configuration Audits

baseline are shown as occurring either late in the concept demonstration/validation phase. However, with the recent trend towards maintaining competition between contractors during development and for programs with increased technical complexity, this review and baseline are sometimes delayed (as shown in Figure 7) until sometime early in full-scale development (perhaps until completion of the PDR). It all depends on the program/product being developed and the acquisition strategy being pursued by the program office.

The overall sequencing of the formal design (technical) reviews is usually contained in the contract master schedule. The breakout of the precise scheduling of the formal design reviews for the system and all component CIs is normally included in the contractor's System Engineering Management Plan which is submitted to, and sometimes approved by, the Government. The determination of the timing and sequencing of the reviews is extremely important and should be given considerable management attention as the contract is being negotiated. If a review is conducted too soon in the configuration item's development, it will be ineffective because the required information about the system or CI will not be available. Conversely, if the design review is held too late, then the design activity is likely to have proceeded further with the design effort even though undetected design errors or shortcomings exist. In that situation, corrective actions could be both difficult and costly. Thus, the program manager must be sure that the design (technical) reviews are scheduled to coincide with the availability of corresponding technical documents (e.g., drawings, specifications, and test plans) so that the contractor's design approach can be assessed at the appropriate time against the stated requirements. Although the design reviews are a systems engineering responsibility, the configuration manager can assist

the systems engineers (either by assuring that the appropriate individuals are present at the review or by providing inputs) to (a) verify that the requirements for the subsystem or CI specified in the documentation to be baselined can be traced to the specification requirements of the higher-level baselines, and (b) validate that the system and CI requirements are/will be fulfilled through the configuration under review.

6.1.1 System Requirements Review (SRR).

A single SRR may be used for the program, or a series of SRRs may be preferred to allow the program office to monitor the development of the system specification(s). The SRR is usually the first type of formal design (technical) review held by the Government and can be conducted either internally within the Government (not the normal approach) or as a joint Government-contractor review. This review may first occur late during the concept exploration/definition phase or early in the concept demonstration/validation phase of the system acquisition life cycle (depending upon the technical complexity or risk associated with the program). The overall objective of the SRR is to determine the adequacy of the contractor's efforts in defining the overall system-level requirements in a system specification and to arrive at a contractual agreement between the Government and the contractor on the systems requirements. This is primarily accomplished by reviewing the results of contractor's system engineering analyses to ensure that the necessary and sufficient system requirements are contained in the system specification (including the delineation of the requirements for the major functional elements of the system, the constraints related to personnel and logistics support, and the verification requirements). The results of the SRR(s) should be a final draft system/system segment specification ready for authentication as the functional baseline.

To facilitate the successful accomplishment of the SRR, the contractor should provide specific documentation and results identified in Appendix A of MIL-STD-1521 [or in MIL-STD-499].

The primary emphasis of the SRR is on the overall weapon system-level requirements, but top-level software functions and requirements must be addressed. If the analyses indicate that a significant computer software development may be involved, then it may be beneficial to conduct a system-level computer software review as a part of the SRR process. The contractor may be required to prepare a System/Segment Design Document (S/SDD) to facilitate such a review. This software review could be accomplished as a separate review, but accomplishing it as an increment of the SRR is probably being the most effective (in terms of cost and performance) in facilitating the overall system's SRR. It also helps in maintaining the traceability to the system review process. This in-depth review would allow for the identification and discussion of possible problem areas and remedial actions relating to the design and development of the proposed computer software elements. The end result of this review would be a finalized S/SDD that would provide the starting point in the allocation of the requirements into the computer software CI specifications.

The purpose of the SRR (or series of SRRs) is to review the system specification in preparation for establishing the functional baseline. Configuration management policy recommends that the functional baseline be established at the beginning of the concept demonstration/validation phase. However, for some major programs, the requirements (and system specification) are not ready to be baselined that early. In that case, follow-on SRRs should be held later in the concept demonstration/validation phase with the end result of establishing the functional baseline. However, the SRR(s)

should focus on the system specification and its contents in order that the mutual agreements about the specified requirements can be reached and a functional baseline established.

6.1.1.1 Concerns of the Configuration Manager. Since the SRR is used to establish the functional baseline for the system, the configuration manager needs to be concerned that the systems engineering process has progressed sufficiently and that the design requirements being addressed at the SRR have adequately captured the overall system requirements. To insure that this has occurred, the configuration manager needs to work with the systems engineers to assist them in performing any required analyses, or to determine that they are performing the required analyses to provide the program manager the confidence that the specification is ready to be established as the functional baseline. Some of the functions that should be included in these analyses are:

1. Establish that there is a clear trace between the operational requirements and the system requirements.

2. Establish that there is a system concept and a functional design, each of which correlates with the system requirements and that the specified system functions are consistent with stated operational requirements.

6.1.2 System Design Review (SDR).

The SDRs are conducted for each hardware CI and are usually held toward the end of concept demonstration/validation or in the first stages of full-scale development. By the time that the SDRs are held, the system requirements will normally have been baselined in the system/system segment specifications. (Where dictated by the

complexity and risk of the system design, the final SRR may be conducted, and the functional baseline established, concurrent with the SDRs for the top-level (prime item) CIs of the system.) The focus of the SDR is the individual CI; it is intended to provide visibility into the allocation and expansion of the system-level requirements into the configuration item's development specification. It will include an assessment of the ability of the CI to fulfill its portion of the system requirements. Normally, the allocated baseline for the CI will be established as a result of its SDR.

Since the SDR addresses a major expansion of the system's functional requirements (into the corresponding CIs), it should include a review of the contractor's understanding of the system requirements as reflected in the functional design approach. The contractor should be prepared to provide the documentation and address the issues included in Appendix B of MIL-STD-1521 [or in MIL-STD-499].

As a result of a successful SDR, the contractor and Government will normally baseline a Type B (Development) Specification for the CI. With authentication of the Type B Specification, the allocated baseline will be established for that CI.

6.1.2.1 <u>Concerns of the Configuration Manager.</u> Since the SDR is used to establish the corresponding allocated baseline for a CI, the configuration manager needs to be concerned that the systems engineering process has progressed to the point that the necessary and sufficient requirements for the CI have been identified from the system requirements and allocated down to the CI. To insure that this has occurred, the configuration manager needs to work with the systems engineers by assisting them in performing any required analyses on the design, or by determining that they are performing the required analyses that will provide the program manager the confidence

that the CI development specification is ready to be established as the allocated baseline. Some of the functions that should be included in these analyses are:

1. Verify the completeness of the CI requirements specified in the development specification, and ensure that there is traceability between the functional and allocated baselines.

2. Validate that the development specification completely and accurately defines the necessary system and hardware CI requirements.

Additionally, the configuration manager should assure that systems engineers address the following issues to determine if the requirements are appropriate for establishment as the allocated baseline. The following list is not all inclusive, but it should serve to focus the review on some important issues.

1. Do the requirements for this CI track to the higher-level CI (or system) requirements.

2. Does the documentation reflect a preliminary design for the CI that clearly identifies the separate hardware and software functions within the CI? [A CI specification should not be established as the allocated baseline until the hardware and software functions have been identified and the requirements for each CI (or CSCI) allocated on this basis.]

3. Similarly, does the allocated baseline clearly specify (such that there are no remaining questions) all hardware/software interfaces for this CI which must be controlled by the Government? [All necessary software and hardware interface requirements should be identified and specified prior to establishing the allocated baseline.]

4. Is there agreement between the contractor and Government personnel concerning the completeness of the specification in defining overall CI requirements? [If either party is unwilling to concur (by formal sign-off) with the specification contents, further work is required to finalize the specification.]

6.1.3 Software Specifications Review (SSR).

The SSRs are conducted for each computer software CI, or they may be conducted for collective groups of CSCIs, depending upon the Government approach to the reviews. The SSR constitutes the final review of a CSCI's requirements prior to establishing the allocated baseline for the CSCI. It is normally held late in the concept demonstration/validation phase or early in full-scale development phase of the system acquisition life cycle after the system-level requirements allocation decisions have been completed and documented in the CSCI's Software Requirements (SRS) and, if needed, Interface Requirements Specification (IRS). In demonstrating the adequacy of the SRS and the IRS(s), the contractor uses the SSR to establish the allocated baseline for the CSCI.

To accomplish the SSR the contractor must provide documentation and be prepared to address the issues included in Appendix C of MIL-STD-1521 [or in MIL-STD-499].

Normally, the SSR for a particular CSCI will be conducted at about the same time in the system acquisition life cycle as the SDR for the hosting hardware CI. It should normally be conducted prior to the start of software top-level design. In some instances, however, the SSR for a CSCI may be delayed until the top-level design is

available. In that case, the SSR will be a sub-meeting within the overall make-up of the PDR for the CSCI.

6.1.3.1 Concerns of the Configuration Manager. Since the SSR is the forum by which the allocated baseline for each computer software CI will be established, the configuration manager should be concerned that the systems engineering process has progressed such that the software functional design being addressed at the SSR adequately reflects the allocated system requirements passed down to the CSCI. The concerns for the configuration manager are similar to those listed in paragraph 6.1.2.1 above. The difference being that, instead of focusing on CIs, the systems engineers and configuration manager should be now focusing on the CSCIs.

6.1.4 Preliminary Design Review (PDR).

The PDRs are normally conducted for each CI in the early part of full-scale development, after preliminary (functional) design efforts have been completed but prior to the start of any concerted detailed design effort by the contractor. A PDR is held for each configuration item to review the complete functional breakout of the CI and to assess its ability to meet the baselined requirements for the CI. The PDR itself may be conducted as a single event (meeting), or it may be held as a series of incremental reviews, depending upon the complexity of the CI. Depending on the needs of the program office, the PDRs may be conducted for each configuration item, or they may be conducted for groupings of functionally related CIs. Regardless of the approach taken for reviewing the preliminary design of the program's CIs, this review should not be conducted for a particular CI until the Type B (Development) Specification has at least been finalized, if not baselined. If the baseline was not

established prior to the PDR, this review should result in the authentication of the appropriate Type B Specification and establishment of the allocated baseline for the CI. [Note: As noted in paragraph 6.1.3, the SSR is supposed to be used to establish the allocated baseline for a CSCI. If this SSR is being held in conjunction with the PDR, then the CSCI's allocated baseline will also be established at this point in time.]

The PDR is the management activity used by the Government to review and evaluate the technical adequacy and the risk resolution of the selected functional design approach. The review of the CI design approach should address the minimization of the technical, cost, and schedule risk. The contractor must demonstrate that the design is ready to proceed with the detailed design phase in that it appears to be able to meet the requirements in the Development Specification.

A complete list of the information that a contractor should have available and the issues that the contractor should be ready to address at the PDR is covered in Appendix D of MIL-STD 1521 [or in MIL-STD-499].

For a CSCI, the contractor will present the developed top-level (functional) design of the CSCI and demonstrate that it is capable of meeting the requirements specified in the Software Requirements (SRS) and, if used for this CSCI, the Interface Requirements Specifications (IRS). In following this approach, the Government reviews the material presented for each CSCI to (a) determine the existence of the interfaces required by the SRS and IRS to be established between the CSCI and all other CIs (hardware and computer software) both internal and external to the CSCI, (b) determine whether the top-level design embodies the SRS and IRS requirements, (c) determine whether the top-level design is a result of the approved design methodology

presented at the SSR, and (d) determine that the test plans establish adequate test criteria for each CSCI and address all specified requirements.

6.1.5 Critical Design Review (CDR).

The purpose of the CDR is to ensure that the detailed design solutions, as reflected in the draft Type C (Product) Specifications and hardware drawings or software design documents, appears to be able to satisfy the performance requirements established by the CI's Type B (Development or Requirements) Specification. This review is normally conducted during the first half of the full-scale development phase of the system acquisition life cycle; a separate CDR is usually scheduled for each configuration item when the detailed design process for that CI is essentially complete. For hardware CIs, this review should be held prior to release of the design for fabrication of qualification test hardware; for CSCIs, this review should be conducted before coding and testing is commenced.

While most CDRs are completed in a single, continuous meeting (i.e. one week long), the CDR for a top-level CI may consist partially of a CDR of the top-level design and partially of an administrative/technical review of the results of the CDRs for the lower-level CIs. In most instances, the CDRs for component CSCIs should precede the CDR for the hosting hardware CIs. This will assure that hardware and software compatibility is maintained.

The correct timing for holding these CDRs is critical. The CDRs for hardware CIs are the final major engineering and technical reviews. The program office must balance the design maturity of the CI, and its ability to meet the specified requirements, against the need to proceed with the testing and the production planning/implementation. Their timing must consider the readiness/acceptability of the

design to proceed with the qualification testing and to commence with many pre-production tasks which must be accomplished in order to prepare for the transition from the fabrication of models and prototypes to the production of operational units. For example, closing out CDRs early can provide more time for production but could reduce the time available to complete all of the analyses needed to finalize the design; this would increase the risk that the number of post-review changes (due to lack of full maturity of design reviewed) will require costly revisions to the production planning. On the other hand, closing out the CDRs too late could delay the start of the qualification testing and may severely constrain production planning schedules and lead to suboptimal manufacturing methods being used.

6.1.5.1 CDR for Hardware CIs. During the CDR for a hardware CI, the detailed design for the configuration item is identified primarily in the engineering drawings and in the draft Type C (Product) Specification. The detailed design reviewed serves as the basis for the contractor's production planning process and for the fabrication of qualification test articles. Therefore, the contractor must place the detailed CI design under internal control no later than completion of the CDR.

To facilitate approving the detailed design, the contractor should provide the documentation, and be prepared to discuss the issues, required by Appendix E of MIL-STD-1521 [or MIL-STD-499]. That documentation is used by the Government to (a) determine that the detail design satisfies the performance and engineering specialty requirements of its development specifications, (b) establish the detail design compatibility of the CI and other CIs and subsystems, (c) assess the risk (technical, cost, and schedule) associated with the selected CI design, (d) assess the producibility of the CI design, and (e) review the preliminary hardware product specifications.

6.1.5.2 CDR for Computer Software CIs. For CSCIs, the CDR is used to evaluate the integrity of the CSCI's logical design prior to its coding and testing. The review focuses on the ability of the CSCI's detailed design to achieve the performance characteristics required by its Type B (Requirements) Specification. The completion of the CDR authorizes the initiation of the source and object code generation. To facilitate the approval of the detailed software design, the contractor should provide the documents, and be prepared to discuss the issues, required by Appendix E of MIL-STD-1521 [or MIL-STD-499].

Since the development of computer software is quite different than that for hardware items, the documentation present at the completion of the CDR for a CSCI is usually not sufficient for the contractor to maintain adequate visibility into the design as it goes through coding, integration, and testing. Thus, at the end of CDR, the contractor must establish an internal control system for the computer software. This will provide the contractor's management the means for precisely directing and controlling the computer software programming process. This design initially released for coding will represent a reference point from which the executable form of the software may be produced and will also provide traceability of the design as it evolves between the formal allocated and product baseline.

6.1.5.2.1 Concerns of the Configuration Manager. Since, at the end of the CDR for a CSCI, the contractor will internally control the design through the Developmental Configuration, the configuration manager needs to work with the program office's systems engineers and the contractor's engineers, by either assisting them in performing any analyses on the design or by determining that they are performing the required analyses, to assure that the CSCI is ready to have the detail design internally

"locked down" so that the CSCI development process can proceed into the unit coding phase. Some of the functions that should be included in these analyses are:

1. Ensure that the design specifies an architectural structure of the computer software which corresponds with the functional and allocated baselines. The design should also reflect a single, unified approach to the computer software design.

2. Ensure that the overall intended function of the CSCI is clearly reflected in the design and that there are well-defined inputs and outputs for each CSCI's design element.

3. Determine technically that the design appears to be able to achieve the requirements specified in the allocated baseline. There should be traceability from that baseline to the Software Design Document.

4. Validate that the CSCI design appears to be able to fulfill its part of the stated operational requirements.

5. Ensure that the Government and contractor concur over the adequacy and suitability of the design.

6.1.6 <u>Test Readiness Review (TRR).</u>

This design review is required for computer software only and is conducted after Computer Software Component (CSC) code integration and informal testing has been completed up to the CSCI level. The TRR allows the program office to determine whether the contractor is ready to undertake formal CSCI testing. The CSCI-level software test procedures will be evaluated to assure their compliance with the software test plans and descriptions and to verify that the test procedures adequately address all of the verification requirements from Section 4 of the SRS. In addition, the program office also uses the TRR to evaluate the technical adequacy of the CSC design from

the results of the informal CSC tests and to determine the adequacy of updated operator, user, and diagnostic manuals (operation and support documents) for the CSCI. To facilitate approval of the CSCI design for testing, the contractor should provide the documents, and be prepared to discuss the issues, required by Appendix F of MIL-STD-1521 [or by MIL-STD-499].

6.1.7 Production Readiness Review (PRR).

PRRs are normally conducted, by the program office, in a time-phased incremental fashion that will usually span the full-scale development phase and encompass the prime contractor (developer and producer if not the same) and all major subsystem suppliers. The objective of these reviews is to determine whether the design is ready for production, whether production engineering problems have been identified and resolved, and whether adequate planning has been accomplished prior to executing a production go-ahead decision. The final PRR represents that point where a production commitment can be made without incurring any unacceptable program risks. Remember that this definition of acceptable risk is dependent upon the acquisition strategy being pursued by the Program Office. For example, if the program office is employing concurrent development and production phases, then its risk level may be somewhat higher than that for a program office that is allows most (or all) of the qualification testing to be accomplished before production is authorized.

The initial PRRs are concerned with generalized manufacturing issues such as preferred manufacturing processes and specialized materials that must be employed to satisfy the desired design requirements. As the CI's detail design matures, the PRRs focus more on the design of the individual CI components to verify that the developer's design is both technically complete and producible. By the end of the PRRs, the

program office will have examined the contractor's production planning documentation, existing and planned facilities, tooling and test equipment, manufacturing methods and controls, material and manpower resources, quality control and assurance, and controls over major subcontractors.

6.2 Configuration Audits.

While the design reviews are the responsibility of the systems engineers, the configuration audits are the responsibility of configuration managers. Configuration audits are used to verify that the characteristics of each hardware and computer software CI meet the requirements of their associated specifications. As the contractor completes (through the systems engineering process) the design and testing of the system and its CIs, the program office must check each CI (and, sometimes, the overall system) for compliance with the baselined (or soon-to-be baselined) configuration identification. The configuration audit process involves the verification of the testing effort and of the detail design documentation. The program office must review the CI's test results to verify that they prove that the CI conforms to the approved functional and physical characteristics for the CI listed in its Type B Specification(s). The program office must also audit to ensure that a deliverable unit of the CI matches the design specified in its product configuration identification (Type C Specification and referenced documentation) prior to the design being baselined and placed under Government control. There are three types of configuration audits that the program office may conduct during the course of the development process, namely the functional configuration audits, functional system audits (formerly called formal qualification reviews), and physical configuration audits.

6.2.1 Functional Configuration Audits (FCAs).

The FCA is normally conducted for each CI as a part of the contracted full-scale development effort and is used to validate that the development of a CI has been completed satisfactorily. This is accomplished by auditing the qualification test results to verify that the CI's actual performance complies with, and has achieved, the hardware/software performance requirements and design constraints defined in the CI's Type B (Development/Requirements) Specification. This audit, and/or the following functional system audit (FSA), is a prerequisite to the acceptance of a CI and/or the total system design. The FCA (but not the FSA) is also a prerequisite for the PCA; it must be accomplished before or concurrent with the PCA for the CI.

The FCA must be conducted on the configuration item (either a prototype or pre-production article) that is representative of the detail design that is to be released for the manufacture of units for the operational inventory. However, if neither a prototype or pre-production article will be produced, then the FCA may be conducted on a first production article. Also, if the configuration item's qualification will depend upon, and be validated by, some type of integrated system testing, then final approval of the FCA for this CI may be delayed until after the integrated testing has been performed (and, if necessary, the functional system audit has been completed). For a complex configuration item, the FCA may be conducted on an incremental basis by auditing the testing results of each major subsystem in the CI or by auditing the results of specific types of tests (e.g., environmental, maneuverability). This incremental approach can be avoided by judicious selection of subsystem CIs early in the development of the CI. If the incremental approach is required, it will normally be finalized by an overall CI-

level FCA that identifies any additional CI-level action items while also finalizing all the action items resulting from the earlier incremental FCAs.

6.2.1.1 <u>Items to be Reviewed.</u> To facilitate the conduct of the FCA, the contractor should provide the data and documentation, and be prepared to discuss the issues, required by Appendix G of MIL-STD-1521 [or by MIL-STD-973, when issued]. The Government will be especially concerned with reviewing:

1. Test plans, procedures, and (especially) test results (listed in test reports) for the CI to verify that it performs as required by its allocated configuration identification. For those CI requirements that may not have been met, the program office must ensure that the proposed course of action (a) will identify and correct the problem and (b) will include adequate testing of new design elements, including repetition of the testing already performed which was invalidated by the redesign. If any of the tests required by the specification have not yet been performed (or are required to be performed as a part of a system-level test program), these must be noted as well.

2. A complete list of the engineering drawings (including revision level) that reflect the CI detail design for which the test data was reviewed and verified.

3. Drawings of parts that have been ordered (provisioned) as spares should be selectively sampled to assure essential manufacturing test date are furnished with the drawing.

4. The effectiveness of the software operating and support documents should be verified by a review of the results of the testing covering their use.

6.2.2 Functional System Audits (FSAs).

Formerly referred to as formal qualification reviews (FQRs), these system-level audits are optional. For most programs once the CI's performance has been verified through the FCAs, the system performance has also been verified, because of the allocation of requirements from the system functional baseline to CIs in their allocated baselines. In cases where a very complex system is involved, an FSA may be required to review the results of the system level tests conducted to verify that the performance requirements specified in the Type A (System/System Segment) Specification have been met. Wherever possible, however, the system-level verifications should be accomplished as a part of FCAs so that system-level problems will be identified prior to the physical configuration audits (PCAs).

Unlike FCAs, FSAs are not a prerequisite for the PCAs. They must be conducted post-FCA (and may be post-PCA) when (a) all system-level performance requirements could not be verified as a part of CI-level testing and the FCAs, or (b) the configuration item requirements could not be adequately verified by the CI-level testing. In many cases, the FSA will be conducted during follow-on operational testing. However, every effort should be made to ensure that duplication of effort with the FCAs is avoided.

6.2.3 Physical Configuration Audits (PCAs).

For computer software CIs, PCAs are often conducted concurrent with the FCA, during full-scale development, using the software that has successfully completed the qualification testing. For hardware CIs, these audits are normally conducted during the production phase of the system acquisition life cycle, typically occurring at the time of delivery of one of the first operational units from the production line. The PCAs are performed by comparing the drawings and other manufacturing documentation, or the

source and object codes, called out in the CI/CSCI product specification against the "as-built" version of the CI/CSCI to be sure that they match before establishing the product baseline using the product specification. The PCA includes a detailed review of the engineering drawings, specifications, software listings and design documents, and other technical/manufacturing data used to produce the configuration item or computer software configuration item.

To facilitate the conduct of the PCA, the contractor should provide the data and documentation, as well as the physical item, and be prepared to address the issues, required by Appendix H of MIL-STD-1521 [or by MIL-STD-973, when issued]. Among other important topics, the PCA will include a review of the effectiveness of the contractor's engineering change release system. The contractor's engineering change release system is reviewed to verify that the contractor has the ability to control changes to the approved detailed design documentation. Once the PCA is successfully completed, any approved changes are processed through the engineering change release system. This review of the engineering release system does not have to be performed during the PCA for each CI; but it must be performed once on the engineering release system at each contractor's facility, usually as a part of the first PCA at that location.

The PCA will also include a review of the contents of the product specification to include a review of the adequacy of the acceptance test requirements and related acceptance test procedures. The performance requirements specified in the CI's product specification are not as comprehensive as the performance requirements in its development specification. However, since they are the basis for the acceptance testing of each unit (or lot) of production deliverables, the program office must be sure

that they are adequate and sufficient. Likewise, it is necessary to verify the adequacy of the acceptance testing procedures (ATPs) to detect discrepant units through quality assurance activities. The PCA unit must be tested according to the approved ATPs, and the results must be carefully reviewed to ensure that the ATPs will allow acceptance of good units and rejection of bad units.

Normally at the completion of the PCA, the CI/CSCI's product configuration identification is established as the product baseline. All follow-on production units will be built to that documented configuration, unless changes are approved by the program office. That initial product configuration must be accurately documented to avoid support problems. Therefore, the program office must emphasize the need for comprehensive planning for, and rigorous accomplishment of, the PCA for each of the program's CI/CSCIs. The "product baseline" for a system is established incrementally through the establishment of the product baselines for its component CIs. In this case, the detail design for the CIs which have completed PCA and had a product baseline established will be subject to the Government (MIL-STD-480) change control process while the detail designs of those CIs still waiting their PCAs will continue to be subject to contractor internal control. (The performance requirements for all CIs are still controlled by the Government under the allocated baseline.) The PCA is normally conducted on the first production article (or on an early production unit), as identified and selected jointly by the program office and contractor. In some cases, a production line may be restarted after a long break, or a new contractor may be selected to produce and deliver the item. In those cases, another PCA may be required; the first unit off the restarted production line or the first unit to be delivered by a new contractor

would be subject to the additional PCA, depending upon the acquisition strategy pursued by the program office (see paragraph 6.2.3.2).

6.2.3.1 Items to be Reviewed. As a part of the planning for the PCAs, the contractor should provide the program office with identification information about the individual CIs and grouped CIs to be audited. This should include each (all) CI's nomenclature, specification identification number, configuration item identifiers, serial numbers, top part/drawing numbers, and computer software/code identification numbers. For each PCA, the Government should review:

1. Authenticated issues of the CI Development Specifications (or CSCI Software Requirements and, if applicable, Interface Requirements Specifications) and all approved changes, deviations, and waivers to these specifications.

2. A comprehensive listing of all design differences between the physical configurations of the selected PCA unit and those "developmental" units used for FCA. Additional test data for differences between these configurations should be provided and reviewed to ensure that the changes do not degrade the functional capabilities of the production design.

3. Final draft version of the CI or CSCI product specification.

4. All approved drawings for the configuration item as invoked through the top drawing number in the product specifications, to be compared to the actual deliverable hardware.

5. Manufacturing instruction sheets to insure they accurately reflect design details contained in the engineering drawings.

6. The contractor's engineering release system and change control procedures to establish its effectiveness in controlling the release of engineering data.

7. Acceptance tests results and procedures for adequacy and for compliance with the product specification performance requirements.

8. Final versions of computer software operating and support documents for traceability to the final software code.

6.2.3.2 Relationships of Audits. According to AFR 14-1, there are three methods by which the FCAs and PCAs may relate.

1. Case One: When the developing contractor is preselected to be the producing contractor then (a) if production of the CI is authorized prior to conducting the FCA, then important functional characteristics of the CI should be selected and "audited" before the production authorization is given, (b) at the appropriate time, a complete FCA of the CI design is conducted, (c) the first production unit or an early production unit most nearly matching the final operational configuration of the CI must be used for the PCA, and (d) if all FCA action items are not closed out prior to PCA, the program office will normally conditionally accept the performance of the production units until the FCA action items are closed out.

2. Case Two: When the developing contractor has not been preselected as the producing contractor, and the production contract is to be competed, then (a) the FCA must be accomplished, as part of the full-scale development contract, and the resulting items closed out prior to accomplishing a PCA of the developmental unit, (b) a complete PCA must be accomplished on the pre-production prototype that most closely approximates the expected production configuration so that the program office can establish a product baseline for future competition, and (c) as part of the production contract, the first (or an early) production unit of a CI from the winning contractor is selected for a complete PCA.

3. Case Three: If at some point after the CI has entered production, a new production contractor (other than the development contractor) is selected to produce the CI, or if an existing production line for the CI has been shut down for an extended period of time, then a PCA may be required to be conducted on the first production unit.

6.2.3.3 Role of the Configuration Manager. Since the technical documentation that is approved at PCA will constitute the product baseline for the CIs/CSCIs, the configuration manager must ensure that:

1. There is traceability between the CI/CSCI(s) requirements listed in the product baseline and the previously established baselines.

2. The product design capabilities correlate to, and fulfill, all stated performance requirements identified in the CI development/requirements and product specifications.

3. For hardware, the actual deliverable design matches the design referenced in the product specification.

4. For CSCIs, the actual deliverable software matches the source listing and data structure contained/referenced in the product specification.

5. For CSCIs, the media (e.g., magnetic disk) which contain the software (data and code) are correctly identified (numbered) and controllable.

6. The contractor and program office concur that the product configuration identification is complete and that the product baseline is ready to be established.

7. CHANGE MANAGEMENT

Change management is the most visible aspect of configuration management within a program office. In addition, it is the principal weapon in the program manager's arsenal of ways to control any increases (or proposed increases) to system costs. As a part of the systems engineering process, the contractor uses an iterative approach to design and develop a product(s) or group of items that meet a stated requirement. At various times during this engineering process, the contractor presents technical documentation (including documents, drawings, schematics, and any updates to previously submitted documentation) at design reviews and/or configuration audits, that describes the product's design capabilities/attributes that have been achieved to date in the system development. At some of the reviews and audits, these documents are then used to establish baselines (that is, they come under Government control). As the system engineering process proceeds, increasingly detailed design documentation is baselined. But what happens to these baselines, or to the approved technical documentation, when changes must be made? How are the contractor and Government able to control (a) the configuration of the product, (b) any changes (requested or approved) to the product and its documentation, or (c) any other aspects of the contract not impacting the baselined requirements?

The means by which this control is obtained, and maintained, is through the application of change management. Change management involves the evaluation, coordination, and decision (acceptance or rejection) of all proposed changes. It is also concerned with monitoring the implementation of any approved changes (in terms of technical or other contractual concerns). Although this monitoring is part of the status accounting process (see Section 8), the program office uses change management to

try to eliminate the submittal of unnecessary or incomplete change proposals (thereby helping to reduce overall program costs), to expedite the approval and implementation of changes deemed to be worthwhile, and to assure proper recording of these changes to help facilitate current and future logistics support of the system/CI.

Change management is composed of two parts - configuration control and change control. This section will first describe configuration control and all its applications in maintaining control over the submitted and approved technical documents constituting the approved configuration identification (baselines) of the program's configuration items. Next, this section will outline the role of change management to facilitate control of the changes to the contract that will not impact the baselined requirements. The section will then go on to provide information pertaining to the Configuration (Change) Control Board, that body of individuals who regulate the change management process for the program office. Finally, the section will briefly discuss a final control concern for the program office, interface control. While primarily a systems engineering function, interface control is used by the program office when two or more contractors for the program have contracts directly with the program office or when two or more Government agencies are involved in the system development. These situations require special activities related to the configuration control process. Configuration managers should be aware of, become involved in, and help maintain control over a program's interface requirements.

7.1 Configuration Control.

Once a program office has formally established the first (normally functional) baseline, procedures must be put in place to regulate the flow of proposals, and to implement any approved changes to the system and configuration item documents

constituting authenticated baselines. The accurate, current, and approved configuration of the system (and CIs) must be known throughout the acquisition life cycle of the system. Configuration control is that part of change management used by configuration and/or program managers to provide regulation of all proposed and approved technical changes. Formal control procedures for the documentation associated with each program baseline must be implemented when that baseline is established. Government configuration control commences when the system specification is approved, authenticated, and placed on contract by the program office; at that time, it must be placed under formal Government control as the program's approved functional baseline. This marks the start of formal configuration control within the program. Similarly, the technical documents (specifications and drawings) associated with the allocated and product baselines are authenticated and placed under formal configuration control at the appropriate times (explained in Section 5) during system/CI development.

DOD policy requires the program office to provide the contractor with the maximum degree of design latitude during the development process. However, as discussed in Section 5, many requirements/constraints and some design interfaces must be baselined and controlled to provide sufficient contractual definition of the requirements for the item development. The program office must assure that all changes to these baselined requirements, no matter how small or seemingly insignificant, are reviewed through an established and well-defined change control process. An effective configuration control process (a) requires the contractor to identify and document the total impact of any proposed change (including to program cost); (b) requires the program office to assure the systematic evaluation, coordination, and timely processing

of the proposed change for approval or disapproval; and (c) assures timely implementation of all approved changes. Thus, configuration control allows the program office to make informed and deliberate approval/disapproval decisions on program changes.

For most programs, configuration control is applied to three types of changes (engineering changes, deviations, and waivers) affecting the Government's interest in the configuration of a baselined system and/or its baselined CIs. All three will be discussed in the following paragraphs.

7.1.1 Engineering Changes.

An engineering change is any alteration in the approved configuration identification of a system or configuration item after formal establishment of a baseline and its corresponding configuration identification. Changes to the documentation which define the baselines are formally controlled by the contractor up to the point in time where this documentation is baselined as a set of contractual requirements for the system or CI. After baselining, any changes to this contractual document will require the formal approval of the program office via the configuration control process.

Configuration control of engineering changes is administered to ensure (a) that formal submittals are minimized through the use of precursor documents, (b) that if unnecessary or incomplete proposals are submitted, they are disapproved, and (c) that only changes which are necessary or offer significant benefits are approved and expeditiously processed. According to MIL-STD-480B, engineering changes are deemed to be necessary and beneficial if they:

1. Correct known, or observed, deficiencies in the system design in order to meet stated requirements.

2. Add or modify interface or interoperability requirements.

3. Make a significant change in the effectiveness of the system operational or logistical support capabilities.

4. Will provide for substantial life cycle cost savings for the program.

5. Will prevent, or allow, any desired slippage in an already approved production schedule.

How are these engineering changes derived, and what configuration control procedures must be in place to ensure that no change will be implemented without prior approval? Ideas leading to engineering changes can be suggested by anyone working on the program for the Government, or for the contractor, who determines that there may be a beneficial change to be made to the system. If the suggestion appears to address a valid problem or opportunity, the remaining steps in the configuration control process will be followed:

1. If the suggested change will affect a contractual baseline, the originator of the desired change must establish the classification of the engineering change as either a Class I or Class II change (see paragraph 7.1.1.1).

2. If the change is classified as a Class I change, and it is further thought to be routine in nature (see paragraph 7.1.1.2.5), then the program office should request the submittal of an advanced change study notice (ACSN) from the originator of the suggestion prior to the formal submittal of an ECP (see paragraph 7.1.1.3).

3. If the change is classified as a Class I change (and if it was routine in nature, its precursor ACSN was approved), or in a select few cases involving Class II changes, the originator must prepare an Engineering Change Proposal (ECP) that describes the change (see paragraph 7.1.1.2).

4. The ECP is then submitted to the Government, (usually the plant representative for Class II and the program office for Class I), for review.

5. If the change is a Class II change, the Government (i.e., the plant representative, if one is assigned for the program) must concur/nonconcur with the classification applied to the engineering change. If the change is a Class I change, the Government (i.e., the program office) must either approve or disapprove the requested engineering change. This Class I decision is normally made by the Configuration (Change) Control Board (see paragraph 7.3).

6. Finally, the approved (or concurred with) engineering change must be incorporated into the configuration item and the appropriate technical documentation.

7.1.1.1 Classes of Engineering Changes. Once it has been decided that an engineering change is required, the originator of the proposed change assigns an appropriate classification. If a disagreement arises over the classification applied, the program office, prime contractor, and originator (if not one of the two mentioned) need to discuss the proposed change and all impacts of the change on the system. The final direction as to the classification to be applied to a proposed engineering change will be provided by the program office, although the contractor is entitled to file a claim against the Government if they continue to disagree.

7.1.1.1.1 Class I Engineering Changes. Class I changes will primarily affect the baselined physical and/or functional interchangeability of either a CI/CSCI or its support elements, and will require program office approval and contractual incorporation prior to their implementation. Engineering changes will be classified as Class I changes

when any one (or more) of the following primary factors are affected [Note: A complete list is given in MIL-STD-480]:

1. Any part of the functional or allocated configuration identification (such that a page change must be issued) is affected to the extent that performance, technical (e.g., weight, balance, moment of inertia, reliability), or interface characteristic requirements would be outside specified limits or specified tolerances. Changes to these identifications are always Class I, regardless of the other impacts of the change.

2. A change to the established product configuration identification that affects the functional or allocated configuration identifications as described above. In addition a change to the established product configuration identification will be considered a Class I change if it impacts one or more of the following:

a. the involvement of Government furnished equipment.

b. safety.

c. deliverable operational, test, or maintenance computer software associated with the CI or CSCI being changed.

d. compatibility or specified interoperability with interfacing CIs/CSCIs, support equipment/software, spares, trainers or training devices/equipment/software.

e. product configuration to the extent that retrofit action is required.

3. There are some non-technical contractual provisions that an engineering change may affect that will also cause the change to be considered as a Class I, even though the remainder of the technical impacts are Class II. These are engineering changes that impact contractor fees and incentives costs incurred by the Government (although many programs establish a Class II dollars threshold to minimize the need

for Class I ECPs merely for this reason), scheduled program/contractual milestones, deliveries schedules, and any changes in contractual guarantees or warranties.

7.1.1.1.2 Class II Engineering Changes. Any engineering change that does not fall into the definition of a Class I change is considered a Class II change. It is important to remember that a change to any of the functional and/or allocated baselines (or corresponding configuration identifications) cannot be a Class II change. A Class II change will only affect the product baseline (and product configuration identification) and then only if there are no functional or physical interchangeability impacts. However, while a Class II engineering change should not affect the "form, fit, or function", it usually will involve changes to engineering or manufacturing data or documentation. Examples of this type of change include: (a) changes in configuration documentation only, such as the correction of typographical errors; correction to software codes that do not affect logic, design, or mathematical formulation; or addition of clarifying notes to the documents, or (b) a minor change (e.g., substitution of an alternative material) to the hardware which will not affect any factor listed for a Class I change.

7.1.1.2 Engineering Change Proposals. Once the technical requirements (either functional and/or allocated) of a system, CI, or CSCI have been baselined, any requested permanent change to these baselined technical requirements, and/or to the corresponding referenced documentation, must be proposed in a Class I Engineering Change Proposal (ECP), normally using the DD Form 1692. This is also true for many requested changes to the product baseline, once it has been established. The ECP forms require the contractor to provide sufficient information to the program office to

completely describe, and justify, all the possible impacts the suggested change will have on all the elements of the system/CI. A single ECP must not cover unrelated engineering changes; rather, if more than one unrelated engineering change idea is being proposed, then a separate ECP should be submitted for each engineering change/idea.

The configuration manager needs to ensure that the originator of the ECP (whether it is the contractor or someone within the Government) prepares the documentation in accordance with either MIL-STD-480 or MIL-STD-481. MIL-STD-480 is the preferred document delineating the configuration control requirements for the approved configuration identification and providing instructions for preparing and submitting ECPs. MIL-STD-480 requires that the design activity provide all the information required with the submitted ECP to describe all performance, interface, training, and support effects, including the exact changes required to be made to the functional, allocated, and/or product configuration identification. MIL-STD-480 should be levied on contracts with the original design activity and on those subsequent production contractors who can reasonably be expected to have the capability of analyzing and providing the Government with the information required on the DD Forms 1692 (see paragraph 7.1.1.2.6). Examples might include a competitive procurement in which the winning contractor (not the development contractor) produces essentially all the operational units. Contractors in a leader/follower or co-production arrangement might also be subjected to MIL-STD-480 requirements.

On the other hand, the Government should not impose an undue burden on the contractor. Thus, MIL-STD-481 should be used to delineate configuration control requirements and provide instructions for the preparation and submittals of ECPs using

the DD Form 1693 when (1) the item is being fabricated to a detail design which was not developed by the current contractor who is therefore not familiar with the item's design and support history, especially when significant quantities of operational units have already been bought from another contractor, or (2) the item being procured was privately developed by the contractor but change control is still required by the Government. When this standard is employed, the responsibility of analyzing the impact of the ECP on associated system elements (e.g., retrofit, technical manuals, spares) is transferred from the contractor to the program office.

7.1.1.2.1 Specification Change Notices. Since configuration control focuses on the contractually controlled/baselined design documentation, any changes to these approved baselines (and their contractually defined configuration identification) will normally require corrections to be made to the content of the approved system/CI/CSCI specifications. [NOTE: Some ECPs to the product configuration identification do not impact the specification content and therefore do not require Specification Change Notices.] Therefore, the originator of a Class I ECP will submit a proposed Specification Change Notice (SCN) as an enclosure to the ECP, although an SCN may not be required for some ECPs which only affect (product baseline) drawings referenced in the Product Specification. The proposed SCN must identify the exact changes to the contents of the specification's sentences, paragraphs, figures, tables, or any other part of its content that must be distributed to holders of the specification if the ECP and SCN are approved. [NOTE: Only the original design activity (the agency whose CAGE identification is on the document) or the Government can use the SCN. Alternate sources must submit a Notice of Revision (see paragraph 7.1.1.2.2) with their Class I ECP.]

The proposed SCN is composed of two parts. The first is the SCN form (DD Form 1696) which is used as a cover sheet and a letter of transmittal, and which includes a list of the current proposed pages to be changed plus a historical record of all prior SCNs and changed pages to this specification revision. The second part of the SCN submittal is the page change(s) associated with the SCN. This page(s), which contains the actual incorporated "wording" changes for that particular specification (paragraphs that are not affected by the change are not changed), is attached to the SCN form and submitted with the ECP. Those portions of the page that are affected are identified by some symbol on the right-hand side of the page. Each changed page is identified by marking the appropriate specification number (including revision letter, if appropriate) and the date of approval of the SCN (date SCN form is signed and returned). The changed page is numbered with the same page number as the page it replaces. If the wording change (addition) is so voluminous that more than one page is required to replace the original specification page, then the additional pages will carry the same page number plus a suffix letter arranged in alphabetical order beginning with the letter "a". In this manner, the changed pages are suitable for direct incorporation into the specification by removing the old and inserting the new pages.

SCNs are assigned numbers in a sequence, beginning with 1, against the original, or current, revision of a specification. Therefore, when a new specification is first authenticated, or whenever a new revision of that specification is authenticated, the SCN numbers begin with 1 again. In addition, once a proposed SCN (required to be printed on colored paper) has been submitted to the program office with an ECP, the number assigned to it will not be changed, altered, or reassigned to any other SCN. SCNs submitted with ECPs may be reviewed and approved in any sequence. For

example, SCN #8 would be proposed before SCN #11, but SCN #8 might not be approved before SCN #11, perhaps due to the complexity of the ECP. Before SCN #8 can be approved and distributed, the "Summary of Previously Approved Changes" section of the SCN must be revised to reflect the SCN #11 information. If this is the case, the number (#8) assigned to the SCN will not change, but the "Summary of Previously Approved Changes" section of the SCN form will have SCN #11 added, and the contents of the change pages proposed with SCN #8 would also change if SCN #11 also affected them.

Once the SCN is approved, the signed SCN form is transmitted from the Government approving office (usually either the program office or the plant representative) to the specification maintenance activity (usually the contractor). That activity then makes the required copies of the SCN/change page package and distributes them according to the Contract Data Requirements List. Each recipient of this latest approved SCN form will insert it into the affected specification immediately in front of Section 1, while the previously approved SCN form in the specification is removed. The change page(s) will be inserted into the specification and the superseded page(s) thrown out.

7.1.1.2.2 Notices of Revision. If the originator of the ECP does not maintain the master (original) specifications, engineering drawings, associated lists, computer software listings, and other technical documents that comprise the approved configuration identification of the CI affected by the proposed change, then the originator must submit a Notice of Revision (NOR) with the ECP. In these cases, the originator of the ECP is not allowed to revise the technical documents nor can they completely document the redesign. If the ECP is approved, the NOR attached to the

ECP allows the program office to direct the contractors who do maintain the technical documents to make specific revisions in the affected documents.

A separate NOR, submitted to the Government using DD Form 1695, is used for each specification, drawing, associated list, computer software listing, or other technical document which will require a revision if the proposed engineering change is approved. The originator of the NOR enters (a) the number of the ECP that describes the engineering change which requires that the technical document, covered by the NOR, be revised, and (b) the assigned system designation (if one exists) and name and type of the configuration item to which the ECP applies [Note: this information should be the same as provided in the appropriate blocks of the ECP].

The contractor preparing the NOR must describe the revision in detail (using a "From" - "To" format), giving exact wording of sentences and paragraphs that are to be revised, or describing exact fields/locations on the technical document that need to be changed. All quantitative requirements (e.g., dimensions, tolerances) that are affected must be addressed. The program office may then (a) tell the originator of a NOR (and/or manufacturer of the affected item if different) that they may proceed, using the existing technical document as modified by the NOR (this requires a copy of the signed, approved NOR to be provided to the originator and to the contractor maintaining the affected technical document); or (b) tell the originator that it is not authorized to incorporate the change proposed by the submitted NOR until it receives a copy of the revised technical document. If the contractor who maintains the technical document was not furnished a distribution list to be used for all changes to the affected CI documentation, then the program office must provide instructions to the

document maintenance contractor with each approved NOR identifying who should receive copies of the revised document.

7.1.1.2.3 Class I ECP Processing. Class I engineering changes may be processed in either one of two basic ways. The originator of the ECP will first submit a precursor document (such as an Advanced Change Study Notice (ACSN) or a preliminary ECP) followed, upon approval of the precursor, by the submittal of a formal ECP, or the originator may submit the formal ECP without using some form of precursor.

The first method allows the submission of preliminary information about the change to the program office without having to incur the expense of generating detailed technical and cost information. This approach should be used when the proposed engineering changes to either the baselined CI/CSCI specifications or other baselined documents are of a routine nature, rather than being urgent or emergency issues, especially when an extensive system engineering analysis will be required as a part of preparation of the ECP. This allows the program office to determine if the proposed change is worth the additional expenditures required to further develop the proposal, or, if the originator is proposing more than one alternative, it allows the program office to select the approach they favor. This method is commonly used with routine ECPs and is strongly recommended once the product baseline has been established. At that point of the program's development, any required change could impact a myriad of support elements, and it will require a considerable amount of money to determine those impacts. Using the ACSN, if the program office decides that the proposed change idea appears to be beneficial, the contractor is directed to generate the additional information necessary to support the formal ECP, and the formal ECP,

complete with SCN and change pages for the appropriate document, is submitted to the program office.

Under the second method, the complete technical and cost information is submitted immediately (without a precursor) as a formal ECP. This ECP submittal provides enough engineering and other program information, in sufficient detail, to support a decision and contractual authorization to proceed. Generally, this type of approach should be used when a change priority of "Urgent" or "Emergency" is involved and expeditious processing of the ECP is required. It might also be used when a proposed change is of a minor nature, such as filling in a "TBD" that is in a specification.

Under either processing approach, a desired engineering change for the basic CI affected may require related engineering changes in other items to maintain an interface match or compatibility. When this is the case, the complete impact against all CIs should be available with the ECP for the basic CI, which means that all related ECPs should be available for simultaneous review and decision. In order to maintain traceability the basic ECP is assigned its number, and all related ECPs are then identified by this number plus a separate dash number. (SCNs for the related ECPs would not use "dash numbers"; they would retain their position in the numerical sequence of SCNs against their CI specification.) The basic ECP will identify and summarize all the related ECPs' cost impact analysis but will not repeat the detailed technical information provided. The program office may then approve/disapprove the basic ECP and all related ECPs simultaneously knowing all the affects their decision will have on system design. [NOTE: MIL-STD-480 cites alternative approaches to be used when the related CIs are not controlled by the ECP preparing activity.]

7.1.1.2.4 Class II ECP Processing. Class II engineering changes usually are not reviewed and/or approved by the program office. For most programs and changes, the cognizant government plant representative will normally review the suggested Class II change and provide concurrence/nonconcurrence with the classification assigned to the change. A concurrence decision by the plant representative tells the contractor that the Government agrees that none of the Class I criteria are impacted by the suggested change. With a nonconcurrence by the plant representative, the contractor can resubmit or request a review by the program office. However, the program office classification decision is final (subject to filing a claim, as noted above in paragraph 7.1.1.1). Examples of cases where the program office might decide on Class II changes include plant representatives or contractors who lack the technical sophistication to properly evaluate the classification of the changes; or a situation where neither the contractor suggesting, nor the plant representative reviewing, the change has custody (or control) of the product's master drawings. Alternatively, the program office may insist on Class II changes being approved rather than using concurrence action. However, this requires a special contractual provision and usually incurs additional costs for the Government. It is therefore recommended that this practice (approving Class II engineering changes) be used only in exceptional cases. Basically, the approval decision is driven by a need to affirm the technical merit of a change as well as its classification.

7.1.1.2.5 ECP Priorities. Each Class I ECP submitted to the program office shall be assigned one of the following priorities by the originator. Unless the program office requests that the contractor change the priority, this assigned priority will be used to

determine the speed at which the proposed ECP is reviewed and decided upon. The priority ranges for Class I ECPs are:

1. Emergency. This type of priority is assigned to an ECP if the proposed engineering change affects operational characteristics which may compromise national security if not implemented without delay. In addition, this type of priority is used to correct a "hazardous condition" which may result in either a fatal or serious injury to personnel or in extensive damage/destruction of equipment. With this type of serious problem, the CI is usually withdrawn from service temporarily, suspended from operation, or development or testing is discontinued until the condition is resolved. MIL-STD-480 sets a goal of 48 hours for the approval/disapproval decision to be communicated to the contractor. Because of the extreme time pressure, the initial submittal is usually a brief preliminary ECP, either via facsimile or electronic message.

2. Urgent. This priority is assigned to an engineering change if: (a) mission effectiveness may be compromised unless the change in an operational characteristic is accomplished quickly, (b) a potentially hazardous condition (one that compromises safety and embodies risk, but within reasonable limits which permit continued use of the affected equipment) needs to be corrected to avoid injury to personnel or damage to equipment, (c) an interface change must be effected to avoid a schedule slippage or increase cost, or (d) the expeditious processing of the proposed change is a major factor in realizing a significant net life cycle savings to the Government. The decision on these changes should normally be officially communicated to the contractor within 30 days of receipt.

3. Routine. If none of the above conditions are met, the ECP is processed as routine. Under this priority, the program office will usually have 90 days to review and decide upon the ECP and officially communicate the decision to the contractor.

7.1.1.2.6 Content and Format. As the configuration identification of the product evolves, the amount of information defined in the baselines and required to be addressed in the ECP increases. An ECP affecting the functional baseline would mainly describe system specification wording changes and qualitative impacts of the change on performance requirements or logistics support. On the other hand, an ECP affecting the product baseline would have to include detailed impact information about changes in part design, changes to production line tooling, retrofit requirements for already delivered units, and specific impacts on logistics supportability (e.g., spares, test software) of the system.

The DD Forms 1692 are normally the mandatory format to be used for proposing all engineering changes. Although the contractor's format might be approved for use in submitting ECPs, use of the DD Forms 1692 standardizes the format and content and minimizes problems with communication about the change. Since word processing software (e.g., Wordperfect) now has the ability to mix graphics (blocks/boxes on a form) with textual material, a contractor's format that essentially matches the DD Forms 1692 is usually acceptable. (It will also allow electronic transmittal of the ECPs, speeding the review/approval process.) Depending upon the life-cycle phase of the CI's development, the submittal must include various pages of the DD Forms 1692 as required by Figure 1 (Life Cycle Applications of DD Form 1692) in MIL-STD-480. The page usage for DD Form 1692 is as follows: [Remember that the ECP is not required until after the first baseline has been established for the system or top-level CI.]

1. ECP DD Form 1692, Page 1, Cover Sheet: This page is required with every Class I engineering change as the cover sheet to summarize the engineering change. This page may be used for Class II engineering changes. However, when only concurrence in classification is required, the contractor's format is normally used for Class II engineering changes, as long as it includes the part number and name of the affected item, the next higher assembly, and a description of the change. However, if the contract provisions call for approval/disapproval of the Class II changes, then Page 1 is required to be submitted to convey the description of the change to be approved.

Page 1, or DD Form 1692, is used to identify (a) the model and type designation of the CI affected; (b) the CAGE code of the contractor submitting the ECP; (c) a system or higher-level CI designation (if there is one and if it is known); (d) the type of ECP (preliminary or formal); and (e) the baseline document affected (e.g., systems/CIs specifications, test plans, and drawings). This page also includes a description of the proposed change (phrased in definitive language such that it could be incorporated into the authorizing contractual document) to include identification of the exact part of the system or CI that would be changed. Also included should be an explanation of the need for the change (e.g., what defect, failure, or malfunction the change is being proposed to correct).

2. ECP DD Form 1692-1, Page 2, Effects on Functional/Allocated Configuration Identification: This page of DD Form 1692 is used to describe proposed changes to the functional or allocated configuration identification as defined by either the appropriate Type A (System/System Segment) Specification and/or Type B (Development or Requirements) Specifications depending upon which baseline(s) is impacted. In addition this page should:

(1) for hardware CIs: (a) qualitatively describe impacts of the proposed change on logistics, personnel, training, and operational/mission effectiveness (employment and deployment), and (b) if the proposed engineering change requires any new prototypes, significant redesign, additional design reviews, or tests to be reaccomplished then the nature of these activities should be described in detail.

(2) for computer software CIs: (a) if applicable, it should identify any required changes to the data base parameters or values and any effects on computer operating and cycle time utilization, and (b) if the ECP is initiated following completion of the preliminary design of the CSCI, then, the originator of the ECP must include specific information to identify significant requirements for computer software program redesign, reassembly, recompiling, recording, retesting, changes to the requirements specification data and source code listings, and any impacts of CSCI changes on the corresponding hardware requirements or on the existing schedules for completion of development.

(3) quantitatively describe any proposed changes on performance parameters contained in the development or requirements specifications. Once the product baseline is established, this page of DD Form 1692 is not required unless: (a) the proposed change to the product configuration identification will impact the system specification(s), (b) the development/requirements specification will require up-dating because of the change in the product configuration identification.

3. ECP DD Form 1692-2, Page 3, Effects on Product Configuration Identification: Prior to the establishment of the product baseline, if prototype test items are undergoing operational evaluation or some form of service tests, then the impacts of changes in the detail design of these existing prototypes which could affect initial

support element planning/orders must be described on this page, even though the Government does not yet control the detail design. Once the product baseline and corresponding configuration identification is established, this page is always required. It is used to describe the details of the effects the engineering change will have on the product configuration identification, on the integrated logistics support elements, on the operational employment of the CI, and other related factors. For the integrated logistic support considerations, this page provides updated logistic support analysis information, spares/manuals/test software which will be required for the support of the new configuration, and/or retrofitting any previously delivered items to the new configuration. For the effects on operational employment, the contractor should look at the quantitative impacts of the change on safety, survivability, reliability, maintainability, and service life. It will also include the details (work-hours per unit to install kits, work-hours to conduct system tests after retrofit, and an estimate of the total time the unit will be out of operational service) of the proposed retrofit for the CI, if any.

For CSCIs, this page will not be used since the information that is required to be reported on this form is either provided on the first two pages or does not apply to computer software programs. Factors normally associated with engineering changes affecting the use and operation of CSCIs depend more directly on the characteristics defined at the Type B (Requirements) Specification than at the product configuration level.

4. ECP DD Form 1692-3, Page 4, Estimated Net Total Cost Impact: Once the product baseline is established, this page is required to tabulate the net cost impacts of the individual ECP for items already on contract or for work that must be added to the current contract. [Savings to the Government are preceded by a minus sign.] All cost/savings (production, retrofit, integrated logistics support, and others) to the

Government resulting from this ECP, if approved, should be shown on this form. However, if approving an ECP for one CI on one program will result in a cost/savings for a change which must also be made for another program/CI, this cost information must be shown on the DD Form 1692-4. If the ECP describes a proposed change that affects items that are delivered to more than one service, then a separate page 4 (DD Form 1692-3) should be filled out for the quantities to be delivered to each service. This page is not normally used with CSCI-only ECPs.

5. ECP DD Form 1692-4, Page 5, Estimated Cost/Savings Summary: This page is required only if there are two or more related ECPs (see paragraph 7.1.1.2.3) or if new trainers or support equipment will be required as a result of the approval of the proposed ECP. This page will summarize and total the net cost impacts (either increases or decreases) of all the individual related ECPs, and adds all related integrated logistics support costs (including maintenance costs) which have not been included on the individual ECPs. The prime contractor is responsible for summarizing all the related ECPs which are its responsibility and, if there is no system integrating contractor, the prime contractor submitting the basic ECP may also be required to include the costs of all related ECPs being submitted by any other affected contractor.

6. ECP DD Form 1692-5, Page 6, Milestone Chart: Once a program enters full-scale development, if the ECP will affect more than just the delivery schedule of the subject item of the ECP, this part of DD Form 1692 is required to summarize in pictorial form the various impacts on other program implementation milestones. Thus, complex scheduling interrelationships, such as the availability of logistics support elements when new/retrofitted production units reach the field, can be more easily analyzed. If the ECP will impact only production delivery schedules, the

analyzed. If the ECP will impact only production delivery schedules, the implementation schedule and interrelationships of the developmental activities can usually be described in detail in the description of change section. Schedules for CSCI and support/training software changes and other developmental milestones for the software should be included on the appropriate lines of the DD Form 1692-5.

7.1.1.2.7 ECP Evaluation and Approval. The evaluation of the ECP should take into account all impacts of the proposed change on all elements in the system. Every proposed engineering change to a system/CI's baseline (and configuration identification) must be evaluated against the alternative of retaining the current design, even if there is a deficiency. In other words, the evaluation of the ECP should include, as an option, not proceeding with the proposed change.

According to AFR 14-1, the program office functionals (i.e., Directorates) should review the proposed change documentation (in their area of expertise) to ensure that all the impacts of the change have been provided. The functionals should evaluate the change to ensure:

1. Adequacy of the proposed ECP for translation into a detailed design that is capable of producing either reliable hardware CIs, CSCIs, or facilities.

2. That the engineering and scientific aspects of the proposed change are effective and current in the light of current research and technology efforts.

3. All interface effects on other parts of the system, subsystems, or facilities have been included.

4. That the effects on system performance, compatibility, item production, and integrated logistics support, if applicable, are included and satisfactory.

5. Specification changes are provided and acceptable.

6. Effects on occupational health, safety, ecology, and the environment have been addressed.

7. The total impact on cost, including all aspects of cost growth and cost saving, have been provided.

Once the ECPs have been reviewed, they are either approved or disapproved by the program office's Configuration (Change) Control Board (see paragraph 7.3) and the decision documented in AFSC Form 318 - CCB Directive (see paragraph 7.3.2). Once the board's documented decision, contained on AFSC Form 318, is provided to the program office's contracting officer, then the decision is provided by the contracting officer to the contractor. Only upon receipt of an official contract modification document, unless otherwise specified by the program office, should the contractor proceed with work on the proposed change. The contractual authorization of the change will reference the appropriate ECP by number, any revision, and date. If the ECP is disapproved, the program office must also notify the originator, in writing, including reasons as to why the change was disapproved.

7.1.1.3 ACSNs. The preparation of formal (fully priced) ECPs and Contract (or Task) Change Proposals (CCP/TCPs) (see paragraph 7.2 below) can be extremely expensive and is almost always charged to the program office. Also, as the number of ECP packages submitted by the contractor increases, the manhours available to review and process each of these changes decreases, and this can also cause a reduction in the manhours available to monitor the basic development effort. The program office should use Advance Change Study Notices (ACSNs) throughout the system acquisition life cycle as a precursor to a submittal of any routine ECP or CCP/TCP. The ACSN (AFSC Form 223) is a useful tool that encourages early communication of a proposed

change idea, reduces the submittal of unacceptable ECPs and CCP/TCPs by validating the need for the change, and avoids the unnecessary costs associated with preparation of these unacceptable change proposals. Its use should be discussed in the contractor's Configuration Management Plan and included as a requirement in the Statement of Work.

In addition to being a tool to reduce the costs associated with change management, the ACSN can also be used to refine the scope of the change effort and to ensure the coordination and integration of all companion or related ECPs and CCP/TCPs prior to beginning the formal preparation of the proposals. The ACSN will provide a brief description of the program deficiency or problem and then provide a brief description of the change idea/solution (and known alternative solutions), identify the program documents affected, and estimate the scope of change effort required. The ACSN may also provide an estimate (called a rough-order of magnitude, ROM) of the proposed change's cost. (In situations where associate contractors are developing various portions of the overall system, the ACSN originator should forward the ACSN to other associate contractors so that their inputs about companion/related changes can be considered as a part of the ACSN review.)

Once the ACSN has been received (including any brief clarifying information resulting from the review), the program office must decide whether it will request the originator/prime contractor to submit a formal change (ECP or CCP) proposal. In deciding that a formal change proposal is required, the program office should ensure that the scope of the desired proposal has been adequately defined for the alternative approach that has been selected and that the proposed change will be beneficial to the program. Conversely, the program office may decide that the change is not beneficial,

inform the originator and all others involved that no further work is required, and thereby avoid the high preparation costs associated with the formal change proposal.

7.1.2 Deviations.

While the ECP is used to document a request for a permanent change to a new preferred design, the prime contractor may also request relief from the preferred design to a less-desirable design for an item. If the contractor determines, prior to the actual assembly of the unit of an item involved, that there will be a discrepancy between the item and the mandatory requirements of the documentation (i.e., the configuration identification), the contractor may request that a deviation be authorized. The contractor must describe in detail the circumstances which have led to the discrepancy and the exact nature of the proposed departure from the technical requirements of the approved configuration identification. The reasons which make it impossible (or unreasonable) to comply with configuration identification within the specified delivery schedule should also be addressed, as well as a proposal of the consideration that will be provided to the Government for accepting the deviation.

7.1.2.1 Designation of Deviations. Section 4 (Quality Assurance Provisions) of the CI product fabrication specification will often contain a Classification of Defects (CD) list identifying certain defects (which may be detected in the components/assemblies of the CI) as critical, major, or minor. In determining the classification of each potential defect, the definitions provided in MIL-STD-109 must be used. If a CD exists, then each request for deviation must be designated by the originator as either minor, major, or critical based on that CD. If a CD does not exist for the CI, MIL-STD-480 provides additional criteria to be used in designating minor, major, and critical deviations.

1. Critical Deviation: if the departure from a characteristic in the documentation involves safety or if it involves a defect classified as critical in the CD.

2. Major Deviation: if the departure involves health, performance, interchangeability, reliability, maintainability, effectiveness in use, weight, or appearance factors, or if it involves a defect classified as major in the CD, then the deviation is classified as major.

3. Minor Deviation: if a CD exists, and if the deviation involves a defect classified as minor in the CD or if it does not fit the departures for critical or major, then it is classified as a minor deviation.

7.1.2.2 Format. The prime contractor may submit a Request for Deviation using either DD Form 1694, a form of its own design, or a letter. If either of the last two methods are used, the following information must be included: (a) name and address of contractor, (b) contract number, (c) name and identifying number of item, (d) number of specification, drawing(s), and/or other document(s) against which the deviation is required, (e) description, and designation, of the deviation, (f) quantity of items involved, (g) if it is a recurring deviation, and (h) effects on contract delivery schedule, and logistics support material. As required by the Federal Acquisition Regulation (FAR), each deviation must include a proposal of the consideration to be provided to the Government if it accepts the item with the deviation.

7.1.2.3 Submittal and Approval. Unless otherwise specified in the contract provisions, critical and major deviations should be submitted in the same manner as a Class I ECP. However, unless very unusual circumstances exist, the contractor should not submit (nor should the program office approve) critical deviations. After review by the

responsible functionals, a decision must be made. Often, this is accomplished by the Configuration (Change) Control Board; at the very least, the CCB chairperson should make the decision as the focal point for controlling the CI's configuration. This decision is then transmitted to the contractor, along with an acceptance or revision of the consideration, by the program office's contracting officer. For a minor deviation, the person having the authority to approve or concur with a Class II change is also normally authorized to approve a minor deviation and accept the consideration.

7.1.3 Waivers.

Whereas deviations are used for discrepancies found prior to the actual manufacture/fabrication/assembly of the unit(s) involved, an item which is found to be discrepant and to not conform to the required configuration identification either during assembly or as a part of acceptance testing requires a waiver to be submitted to the program office before the unit(s) can be accepted. In addition, the contractor should describe what consideration will be provided to the Government if it accepts the Request for Waiver.

7.1.3.1 Designation of Waivers. Each waiver must be designated as minor, major, or critical based on the impact of the defect involved. As with deviations, if a classification of defects is established for the CI, the definitions of MIL-STD-109 are used to determine classify the defects. The contract may also specify an acceptable quality level (AQL) for the acceptance or rejection of a lot of items based on the number of defects in the sample. MIL-STD-480 also provides other criteria besides AQLs and CDs that can be used for designating the waiver as minor, major, or critical.

1. Critical Waiver: need to accept a nonconforming item involving safety or involving a defect classified as critical.

2. Major Waiver: need to accept a nonconforming item involving either (a) health, (b) performance, effectivity in use, weight, reliability, maintainability, appearance, or interchangeability factors or involving a defect classified as major.

3. Minor Waiver: need to accept a nonconforming item which involves none of the factors for major or critical waivers, or involves a defect classified as minor.

7.1.3.2 Format. The contractor may submit a Request for Waiver using either the DD Form 1694, a form of their own design, or a letter. If either of the last two methods are used the contractor must submit the information listed in paragraph 7.1.2.2 above. In addition, the waiver must include the identifying number of the inspection or test plan which revealed the defect, the pre-delivery corrective action (repair) required, and a proposal of the consideration to be provided to the Government if the item is accepted as is or with some repairs.

7.1.3.3 Submittal and Approval. Unless unusual circumstances exist, the contractor should not submit critical waivers to the program office. Minor waivers should normally be submitted to the contractor's Material Review Board (MRB) for disposition. The program office or plant representative will normally have reviewed the contractor's MRB to ensure it is properly constituted and maintained. If so, then the MRB is generally authorized to approve or disapprove the minor waiver. However, a major waiver should normally be submitted in the same manner as a Class I ECP. After review by the responsible functionals, a decision must be made. Often, this is accomplished by the Configuration (Change) Control Board; at the very least, the CCB

chairperson should make the decision as the focal point for controlling the CI's configuration. The decision is then transmitted to the contractor, along with an acceptance or revision of the consideration, by the program office's contracting officer.

7.2 Change Control.

The second aspect of change management is change control. Change control involves controlling and maintaining, as well as approving and implementing, any proposed changes to contractual requirements (documents) that do not impact the baseline requirements (e.g., changes that do not require the revision of approved configuration identification). The documents used for change management are called Contract (Task) Change Proposals (CCP/TCPs).

CCP/TCPs are primarily written against the requirements listed in the Statement of Work (SOW) tasks, in contractually required delivered plans (e.g., test plans, System Engineering Management Plan, and Software Development Plan), and contract data requirements list (CDRLs). The requirements for processing CCP/TCPs must be included in the SOW and the CDRL if these change control documents are to be used. Likewise, the program office must plan for reviewing and approving the CCP/TCPs following procedures similar to the ones for Class I ECPs, including use of the Configuration (Change) Control Board to make the decisions.

While the Data Item Description (DI-A-3020) for the CCP/TCP provides a standard format that the program office can require the contractor to use to submit a CCP/TCP, the exact details of the proposal's content, and the kind of detailed information required to be submitted to the program office, needs to be worked out with the contractor. The CCP/TCP form does provide some general guidance about the type of information that should be included: (a) contractual documents affected (e.g., SOW, test plan, etc), (b)

benefit of making, and impact of not making, the proposed change, (c) description of the proposed change to include detailed description of the tasks involved (with details of man-hours and special equipment required), (d) any alternatives to the proposed change, including reasons for and against each and the estimated cost, (e) detailed cost information for the tasks involved, (f) schedule for completing the work, and (g) the priority of the change and the related decision need date.

As with ECPs, the cost associated with preparing and submitting formal CCP/TCPs may be quite high. Therefore, the program office may request that the contractor use an ACSN, as discussed in paragraph 7.1.1.3 above, as a precursor to routine CCP/TCPs. If the ACSN is approved, the CCP/TCP is submitted, reviewed, decided upon at the CCB, and a CCB Directive is provided to the contracting officer who proceeds to negotiate a contract modification to authorize the contractor to proceed.

7.3 Configuration (Change) Control Board.

Formal configuration control begins with the establishment of the functional baseline and continues throughout the acquisition life cycle for a system and its CI/CSCIs. Change control needs to be implemented immediately after the contract(s) have been signed by the contractor and the Government. For both change control and configuration control, there must be some focal point for making decisions on the proposed changes.

The primary agency established by the program office to regulate the change management process, and to be responsible for change decisions, is the Configuration (Change) Control Board (CCB). The CCB is established at the system or at the top CI/CSCI level, usually early in concept demonstration/validation, around the time of the baselining of the functional configuration identification. Official administrative orders

are issued designating the membership of the CCB for the program and their roles. The CCB receives a summary briefing about each change, deliberates about the major impacts of the change and its implications, and provides advice to the chairperson who makes the final decision about the change. While the program management responsibility rests with the developing command (normally AFSC), the program office's CCB is the agency with official authority to act upon, and provide a disposition for, each proposed change document (including ECPs, deviations, waivers, ACSNs, and CCP/TCPs) submitted against the program.

The program manager responsible for the acquisition of the system is normally the chairperson of the CCB. The chairperson is responsible for the final decision on all proposed engineering and contract changes. In most instances, especially for larger programs, the program manager needs to concentrate on other program functions. In that case, the program manager will delegate the CCB chairmanship to the deputy program manager or to a trusted functional manager; this delegation will be accomplished through the CCB orders. The configuration manager will normally be designated on the orders as the CCB's secretariat, to serve as the secretary at all CCB meetings. The secretariat duties require the configuration manager to be responsible for scheduling, administering, and documenting the results of all CCBs.

Other members designated on the CCB orders will normally include the heads of the directorates (or divisions) for each functional activity within the program office (e.g., engineering, logistics, manufacturing, test, contracting, configuration management) and representatives from the training, supporting, and using commands associated with the program. Alternate members are also included on the orders to allow for absences (e.g., leave, TDY) of the primary members. These individuals must represent their

respective organizations and present their organization's official position on all matters brought before the CCB. The CCB orders should be reviewed, updated, or revised periodically (AFR 14-1 requires the orders to be revised at least every six months) to reflect and maintain current membership.

Depending on the type of change being proposed, the program office CCB may request support and/or advice from other sources such as Interface Control Working Groups, software configuration sub-board, technical specialists, not-for-profit contractors, or other Government agencies. The contractor (or subcontractors) may sometimes be requested to attend a CCB meeting, as a non-member, to provide additional information/insight into a particular area of review.

In addition to these functions, since the CCB has the responsibility for <u>controlling</u> the baselines, some programs may want to use it to <u>establish</u> the baselines. This could be accomplished by having the prime contractor submit the final draft specification as an attachment to an ECP for authentication and contractual incorporation. The submittal would occur after the program personnel have had an opportunity to review, comment on, and negotiate the contents of the draft specification. The CCB would review and approve the ECP and provide direction to authenticate and contractually incorporate the specification.

7.3.1 <u>Preparing for the CCB.</u>

The configuration management directorate is responsible to the program manager to establish and maintain all change control procedures for the program office, to be an active participant on the CCB, and to provide a secretariat for the CCB. When a change proposal is received (whether it is in the form of an ECP, ACSN, deviation, waiver, or CCP/TCP), the configuration management directorate will usually assign it to

a specific configuration manager who will ensure the change proposal is recorded and tracked as it progresses through the stages of review, coordination, deliberation, approval/disapproval, contract authorization, and implementation. For this reason, the configuration management directorate may want to design a "Change Log" that could be used to give a real time status of, and historical information about, all changes being, or which already have been, processed through the program office. However, it should be recognized that, while the persons handling the changes are providing and using the information, they are performing a status accounting function (see Section 8).

When a change proposal is received at the program office, the configuration management directorate (or that unit within the program office performing this function) should perform a preliminary review of the proposal. To maintain tight change management (change and/or configuration control), the configuration manager assigned should review the submittal to insure that it is authorized (e.g., being submitted against a baselined or non-technical document that has been placed under contractual control); that it has a justifiable priority listed (this establishes target processing times and suspense dates for coordination and decision); that it includes the necessary and essential information to allow for an adequate review of the technical, contractual, and cost impacts to the program; and that the proposal includes a reasonable need date for implementation. If the proposal seems to be lacking in any of these areas, the configuration manager should discuss the deficiency of the proposal with the contractor. If these proposal deficiencies are not resolved through this informal correspondence and discussions with the contractor, the change proposal should be returned to the contractor, through the contracting officer, complete with written notice of the deficiencies. If these discrepancies can be, or are being, resolved

without the need for a revised proposal, then the proposal can be further processed so as not to delay the boarding of the change unreasonably.

After the proposal is "logged-in" by the configuration manager, it should be formally distributed for coordination to all the functional directorates of the program office and all the other CCB-member agencies involved in the program, soliciting their position and comments on the proposed change. (Since the contractor often is required by the CDRL to send the change proposal document directly to all the addresses on the distribution list, the coordination letter is often sent out by itself referencing the proposal document.) The coordination letter requests that the addresses provide any comments about, and their organization's official position on (either acceptance or rejection), the change to the configuration manager by a specific date. Copies of the comments will also be provided to the appropriate project officer handling this change. If comments received involve corrections or changes to the proposal content, the contractor's concurrence must (per AFR 14-1) be obtained, either verbally or preferably in writing, prior to boarding the proposal incorporating the changes. As a part of the coordination letter, the configuration manager should provide a schedule of when (date, time, and location) the proposed change will be discussed at the upcoming CCB and the related date by which the comments should be provided.

Each functional activity should evaluate the change proposal to determine if the change affects: [Notes: The following list is not all inclusive, but gives the configuration manager and other functionals a point for initial departure. In addition, not every member of the CCB will be able to answer all of the following, but should "zero-in" on those areas of their expertise.]

1. The performance of the system or CI/CSCI listed, and if there are any other CI/CSCIs in the system that may be affected?

2. Any CIs that are used by other systems?

3. Any interface requirements external to the change originator's design? If so, is there an Interface Control Working Group established that the CCB may have to interact with over the proposed change?

4. Manufacturing tooling or software? If affected, can the change be implemented by the contractor's manufacturing division as scheduled?

5. Drawings, specifications, software manuals, and/or test procedures? If so, has the contractor provided the proposed changes with the proposal?

6. Delivery schedule, training devices, support equipment, testing (will some need redone or will new additional tests need to be performed)?

7. Total program cost? If so, has the contractor provided complete cost information about current contracts affected and supplementary life cycle cost estimates (if required) for future costs/savings related to the change?

8. Stocks of parts for the production line and in the field? Will those in-plant have to be scrapped? Will those that have already been delivered need to be returned for rework or retrofit?

9. Operational, test, or maintenance computer software? Will other computer software (e.g., digital manuals) need to be modified?

7.3.2 Running The CCB.

When a proposed change (whether it is an ECP, deviation, waiver, or CCP) is presented to the CCB, a single individual (commonly a project officer or the configuration manager) will present the change to the board members. The briefer

should assure that functional (e.g., cost, engineering, logistics) experts are in attendance to answer detailed questions; in some cases those experts will brief their part of the change. The briefing should include all the information the chairperson needs to understand the change and to make a decision; pre-coordinating with the chairperson about a standard briefing content/format will simplify the conduct of the CCB. In general, however, the presentation should include at least (a) a discussion of the CI/CSCI and its configuration identification affected, (b) why the change is needed, (c) what are the significant impacts to the program if the change is approved? If it is disapproved?, (d) cost and price of the proposed change, if known, or a rough-order of magnitude if the actual cost is not known, (e) pre-CCB comments and coordination positions (for or against) by each of the responding organizations, and (f) the recommendation of the presenter on the proposed change. After the presentation, the chairperson may discuss the change and ask questions of the CCB members and functional experts in order to understand the change's importance/benefit to the program. During the discussions and decision making of the CCB, the configuration manager (or appointed assistant) must maintain adequate minutes of the proceedings. While they don't have to be a verbatim record of all discussions, the minutes should include at least the key points of the discussion for each change proposal, the CCB decision, and the identification of the affected CI/CSCI(s).

When the presentation and discussions are over, the board may then decide to: (1) approve the change as written, (2) disapprove, with the appropriate reasons provided, (3) approve the proposed changes, but with specific comments or adjustments to the proposal as written (NOTE: If any additional changes are required they must also be coordinated with the contractor before the CCBD is sent to the

contracting officer], (4) defer any action on the proposal until some further information, usually requested in the form of an additional study, is acquired (the board will assign a specific due date for this information to be provided), or (5) refer the proposal to the next higher authority, with appropriate recommendations and comments, when approval is outside the jurisdiction of the program manager. Whichever decision is made, it should be documented on a CCB Directive (CCBD) (AFSC Form 318), or an equivalent form, which becomes the official record of the board's decision. The CCBD will include a record of the concurrence/nonconcurrence of each board member with the decision of the CCB chairperson. Additionally, the CCBD should contain a suggested implementation date for the approved change (if it is different from the "need date" specified in the proposal) and may include a recommendation for the use of contract change order, if immediate implementation is required. The chairperson's signature establishes the CCBD as the official direction for the proposal.

After assuring that the CCBD is adequately filled in (including the chairperson's decision and the other CCB members' concurrence or nonconcurrence), the configuration manager becomes responsible for the distribution of the CCBD to the contracting officer and the other involved agencies (but NOT the contractor) to serve as official notification of the board's decision for a proposal. For the other involved agencies, the CCBD provides the program office's approval decisions to those agencies; as a result, those agencies will undertake whatever action may be required of them for the change to be completely implemented.

7.3.3 Contractually Implementing the CCB Decision.

For the contracting officer, the CCBD provides direction to prepare and issue a contractual instrument to implement the change. The contracting officer is not allowed

to alter the decision of the CCB as documented on the CCBD unless an official amended CCBD is received. The contractor is not authorized to implement an approved change without first receiving official contractual authorization from the program's contracting officer. In most instances, this contractual authorization will be in the form of a negotiated bilateral, fully-priced contract modification (termed a supplemental agreement) between the Government and the contractor. It will include the fully negotiated price adjustments to the affected contracts. Prior to the issuance of this authorization, the program office will conduct a fact finding investigation and detailed negotiations about the required work effort for the proposed change. This investigation is performed to review the contractor's proposed work tasks and the associated costs, to insure that the proposed price, work tasks, and the time required to perform the work as stated in the proposal is adequate and necessary.

In some cases, the approved change is urgent and requires more expeditious implementation than is possible using a supplemental agreement. In order to provide the contractor with almost immediate authorization to proceed, the program office may issue a unilateral, unpriced, contract modification (termed a change order) which will include a not-to-exceed (NTE) cost limiting the Government's monetary liability until a supplemental agreement is issued. The use of the change order is subject to considerable scrutiny and control, so it should only be used when it is critical for the work to commence in order to minimize delays in the program schedule and/or to minimize cost growth.

7.3.4 Maintaining Change Files.

As the program office's focal point for change management, the configuration management directorate must maintain accurate and precise records for all change

proposals. In addition to the "logging-in" of each proposal, another successful approach is to maintain a permanent, separate file for each proposal received. This file should record all the activities associated with the proposed change. The file should contain at least a copy of: (1) the precursor document (e.g., ACSN), if one was submitted; (2) the change proposal; (3) the responses from all coordinating activities required to review the change; (4) the minutes from each CCB that discussed the change; (5) the CCBD including concurrence/nonconcurrence of board members; and (6) the contractual instrument (change order and/or supplemental agreement) used to implement the change.

7.4 Interface Control.

Interface control is the final area of control that the program office needs to be involved in during a product's development. Interface definition and control are an integral part of the systems engineering effort. In addition to identifying the system/CI functional elements and performance allocations, system engineering is concerned with the documentation and control of all physical and functional interfaces of the system, equipment (including support equipment), computer software, and facilities. Since the contractor usually conducts the systems engineering effort, the interface definition, generation of the interface documentation, and control of those interfaces is primarily a contractor responsibility. However, some of those interfaces will be controlled by the Government through requirements specified in System, System Segment, Prime Item Development, Software Requirements, and Interface Requirements Specifications. Control of those specified interfaces is exercised through configuration control/ECPs as the interface requirements are baselined.

For some programs with very complex interfaces, the program office may want to implement more in-depth interface control procedures. This might be needed if the program office feels that the interfaces of the affected items (developed by different contractors and/or controlled by different program offices) need to be precisely defined and controlled to assure successful product development. In this case, the program office would use the same specifications to define detailed interface (functional and/or physical) requirements to a deeper level within the CI/CSCIs. The program office will specify these critical interface requirements (by referencing the interface control drawings/documents developed by the affected contractor(s)) in the system/system segment specifications and development (Type B) specifications; in appendices to these specifications; or as in the case for CSCIs, in the software requirements or sometimes in a separate interface specification. These interface requirements/ documents are placed under Government control as the related baselines are established and require the submittal of engineering change proposals to make any alterations to an already baselined interface. The risk with this approach is that many details of the design are being "locked down" very early in the development process and that many times (in fact in almost every case) the evolving design will continue to change as it progresses towards its product baseline. Every time a change to a controlled interface is required, the interface change must go through the formal ECP process. This can result in longer schedule times and higher program costs.

The following paragraphs will address interface control procedures that may be required to supplement the contractor's, and Government's, formal interface (baseline) control practices. In cases where complex contractual or management arrangements affect the system or item being developed (such as when two or more Government

(associate) contractors, or Government agencies, are involved in developing and managing items whose individual configurations may affect/impact one another), specialized interface control practices may be required to ensure that the system/item's characteristics that interface with related systems, components, and support equipment (hardware and/or computer software) are carefully defined, monitored, and informally controlled during the system (and CI) development. Interface control is that part of the system design effort that will both generate and administer technical agreements between two or more affected agencies. It requires a technical effort to arrive at these agreements and supporting documents, and a continuing technical/administrative effort to control the derived agreements and documents.

Although interface control is considered a part of the systems engineering process, the program office's CM personnel should be knowledgeable of the interface control process being used during development. Configuration management personnel will be very familiar with the configuration control of high-level interface requirements in the specifications. However, in order to promote design flexibility for the contractor, program offices normally delay formal Government control of the lower-level interfaces (functional and physical) until the establishment of a CI/CSCI's product baseline and product configuration identification. During development, the contractor is responsible for defining, controlling, and integrating all lower-level interfaces being designed by its personnel or by its vendors and subcontractors.

For some very large systems, however, the Government decides to contract directly with designers of major pieces of the system. They are called associate contractors. In this case, the prime contractor no longer deals directly (contractually) with these associates, and the Government usually then tasks (and pays) the

associate contractors to establish management (including interface) communications. Under this approach, the Government tasks the prime contractor to establish an Interface Control Working Group (ICWG) (see paragraph 7.4.2 below) and prepare interface control drawings/documents (ICDs) (see paragraph 7.4.1 below) to define and control the proposed interfaces. The ICD defines an interface agreement established between the participants by the ICWG as to the lower-level interface requirements for the system/CIs. With the exceptions noted earlier in the section, the ICD(s) is not subject to Government control during development. Similar procedures may be used among multiple Government agencies involved in the development of a new item/system. This type of ICWG would define and control the interfaces between the item under development and the using (hosting) systems controlled by the agencies.

If an associate contractor ICWG is involved, the program office must maintain some awareness of what is happening with those lower-level interfaces. A Government representative will normally attend the ICWG meetings as an observer. The individuals responsible for monitoring and managing the contractor's systems engineering effort will normally be the program office's focal points for this ICWG and ICD activity to ensure that the appropriate procedures are being followed and that the Government-controlled interface requirements are not being affected.

As long as the contractors (or the other agencies) involved in the interface development agree on the characteristics required of each for the interface, and the definition is otherwise consistent with all contractual interface requirements, the program office allows the ICWG to control the ICD and implement any changes. The advantage is that no contract changes are required to revise the ICD (so delays and higher costs are avoided), and the contractors are given the flexibility to design a

system that meets the specified interface requirements. However, once the final production design has been generated and the item's product configuration identification is ready to be baselined, the program office implements formal change control procedures over the detailed design, including all the interfaces.

7.4.1 Interface Control Drawing/Document (ICD).

According to DOD-STD-100, ICDs are used to depict the functional and/or physical interface requirements of a configuration item or component which must be addressed as a part of the design and/or operation of both this item and some other item(s). ICDs are often referenced in specifications as the means of contractually defining the interface requirements. ICDs are also used by the ICWG as a means of recording design agreements among all the participants. They provide a means to measure, evaluate, and informally control the interface design parameters (hardware, computer software, and equipment) required of each participant. As design control documents, ICDs ensure that interface characteristics, and requirements, are delineated such that (a) compatibility of all affected items is established and maintained, (b) control is established to prevent changes to requirements that would affect the compatibility of these items, and (c) design decisions and changes are communicated to all participants.

7.4.2 Interface Control Working Group (ICWG).

If more than one associate contractor or Government agency is involved in the development of items whose individual configurations may impact one another and thereby affect system performance, an ICWG may be established. The ICWG will serve as the official communications link among the participants and will provide them

with the means to establish and document the agreement on interfaces among the items, resolve any interface problems that arise, and coordinate all engineering changes to the items that will impact the interface requirements. The members of the associate contractor ICWG should include at least one member from each associate contractor plus an observer from the Government. The members of an intergovernmental agency ICWG should include at least one member from each Governmental agency plus an observer from the contractor. The members must have been given the approval authority to commit their respective organizations to technical agreements. The chairperson of the ICWG, established through some type of formal agreement, is provided from either the program office, prime contractor, or integrating contractor (if one is involved in the program). The chairperson, or appointed alternate, is responsible for preparing an agenda for each meeting and ensuring that minutes of the proceedings are kept. Similar to the case with the program office's CCB, a roster of the members and their alternates may be officially recorded in a contractual letter or order.

7.4.2.1 <u>ICD Processing.</u> Any member contractor who sees a need to establish or modify an interface may submit a draft ICD to the ICWG. The ICWG chairperson, or secretary, distributes the ICD to all affected parties (contractors and Government agency) for review and requests that they submit their recommendations to the ICWG chairperson. After all the members have reviewed the proposal, the ICD will be considered at the next scheduled ICWG meeting and either approved or disapproved. If an informally controlled ICD is approved, the change is implemented. However, if the interface or ICD involved is under Government control, it is passed on to the program office for review and direction is given to prepare an ECP. If the ICWG

unanimously disapproves the ICD, then it is returned, with comments, to the originator. If disagreement exists within the ICWG, such that the ICD cannot be either unanimously approved or disapproved, the ICD is submitted by the ICWG chairperson (with recommendations) to the program office for resolution.

7.4.2.2 Need For Engineering Change Proposals. When the ICWG (or program office, if a unanimous decision has not been reached within the ICWG) approves a change to an interface or ICD that is under Government control, an ECP must be generated. Once the program office decides an ECP is needed, the ICD is returned to the initiator with direction that an ECP should be prepared. (In this application, the ICD serves the purpose of an ACSN.) If more than one contractor's interface is affected, each affected contractor must be directed to prepare and submit a related ECP. This ECP is then submitted, with the ICD, to the program office and to the ICWG members so that all member contractors may review the proposal. Upon their final review, the members provide their recommendations to the ICWG, which provides a group recommendation to the ICWG chairperson. The ICWG chairperson will then pass on to the program office's CCB, the recommendation (approval or disapproval) of the ICWG on the ECP. The program office CCB will then make their decision as to either approve or disapprove the ECP.

8. CONFIGURATION STATUS ACCOUNTING

From the first three functions of configuration management, the program office is able to establish the technical baselines and related functional, allocated, and product configuration identifications; to verify and validate the development effort and technical documentation of the resultant design; and to ensure that procedures are established for the review and approval/disapproval of any proposed change (to a baseline or contract document) prior to the change's implementation. These three CM processes (configuration identification, configuration audits, and change management) provide the means by which the rigor and integrity of the system's technical development is maintained throughout its acquisition life cycle. However, to successfully implement and maintain these processes, considerable amounts of management information must be available. Otherwise, how does the program office know where it has been, where it is, and how it is doing in the system development or the system operation? The answers are provided through the implementation of the last CM process available to a product's program and/or configuration managers. This process is referred to as configuration status accounting.

This section will discuss the aspects of the configuration status accounting process. First, the notion of status accounting as a management information system will be addressed. This is followed by a brief outline of the status accounting responsibilities for each of the participants involved in the product's development. The relationship, and interweaving, of the configuration status accounting functions to the other three CM processes is discussed. This section will also address the role, and importance, of configuration status accounting as the product progresses through the system

acquisition life cycle. Finally, ideas of the types of logical groupings of information required for certain important management functions will be addressed and presented.

8.1 <u>As A Management Information System.</u>

Configuration status accounting (CSA), in its simplest form, is a management information system that establishes and provides a configuration-related information data base that assures a lifelong traceability of the system, its configuration items, and their related documentation. The management information system should be flexible, be able to meet stated requirements, be tailored to the program's size and needs, and, if possible, be provided using existing contractor and Government systems. It should tie together the other CM functions (processes) and provide both current and historical information to the program office, the proposed using agency, the supporting agency, the contractor, and any other Governmental agency involved in the system's development or operational use. Configuration status accounting provides the traceability of the activities resulting from the other three CM processes. Configuration status accounting is intertwined with these other CM functions/processes. It is the way by which the outputs from these other processes are recorded, maintained in a data base, and accessed for various management purposes.

The CSA system must record (either manually or using some automated method) information about the configuration identification documentation, about the identification numbers for the CIs and their component elements, about proposed changes being considered, about approved changes being implemented, and about the configuration(s) of delivered units. It may also be used to record information about events of significance (e.g., establishment of a baseline, approval of a FCA minutes) concerning the system's development. Such a record should include a description of

the event and the date of the event. The details of the information to be tracked by the contractor should be agreed upon by the contractor and program office and included in the contractor's prepared Configuration Management Plan (CMP). As the configuration manager in the program office, if there is any doubt as to whether any particular piece of information should be stored or not, capture it and record it in the data base.

In addition to the CSA information provided/stored by the contractor, there is also CSA information that needs to be provided/stored by Government participants. The CSA system must be able to record/store contractual information (usually provided by the program office) about each CI, primarily the contract numbers and number of units on order with each contract. The CSA system must also include information (usually provided by the program office) about the processing of changes to the CI and keep track of all activity associated with the change (e.g., processing events, CCB decisions) until it is disapproved or incorporated into the contract. Finally, the Government (normally AFLC or its responsible Air Logistic Center) should also provide detailed information about the purchasing of spares and the status of maintenance and retrofit changes for field unit configurations.

In its data maintenance function, CSA should organize and store everything that was recorded, including updates to that information, in an indexed, easily accessible location. It must provide for a complete and organized data base of all required information that is captured during the system development/acquisition life cycle.

Finally, as a management tool, CSA will provide the focus for communication about the technical status of the program among the program office, contractor(s), and supporting and using agencies. The data base will be used to generate scheduled CSA submittals or will provide the source data used in an interactive customer

information system. (The system must include guides/keys to help users interpret and understand the information being viewed.) Additionally, the data base provides a history of the development process as a basis for post-development analysis and for future project estimates. To provide this enhancement to the management of the program, status accounting should be recording:

1. Configuration identification information such as name, number, revision, and issue date of each specification, drawing, and computer software listing that is part of the CI's various baselines. Additionally, the date that each CI baseline was established should also be recorded.

2. Configuration item data such as the nomenclature, serial number, part number, and any other identifying number of each CI and that of the next higher level CI, if there is one, of which each CI is a part.

3. The receipt and timely processing of change proposals. Status accounting should be recording the CI nomenclature, title, number, and the priority of the change submitted. The proposal processing route is identified, with suspense dates for required responses recorded.

4. Information to identify and monitor the implementation of all approved changes to the configuration of the system and its parts. This requires the change itself to be identifiable by type of change (technical or contractual) and affected contract, the configuration identification documentation affected to be recorded, and identification of any changes to the configuration items themselves.

5. The actual delivered configuration of operational units and changes due to maintenance and/or retrofit activity.

While we are concerned about CSA functions in this section, many of the CSA functions are integral to the other three CM functions. For example, tracking current revision/SCN levels of a specification is often thought to be a configuration identification function, and it would probably be accomplished by that functional group, but it is basically a CSA function. Likewise, tracking the status of an ECP under review is a CSA function, not configuration control. Configuration status accounting provides the information data base that allows the other CM functions to operate effectively. However, it must be responsive to the information needs of all users (not just CM users), both in terms of standard data/data base analysis products immediately available to the user and in terms of rapid response to "ad hoc" requirements (provided in response to a special inquiry by one of the project participants).

8.2 Relationships With the Other CM Functions.

Configuration status accounting is the management information system that records and reports the results of the configuration identification process, the results of the design reviews and configuration audits process, and the results of the change management process for each configuration item and its component parts.

8.2.1 Relationship With Configuration Identification.

The CSA process must start with the establishment of the functional baseline. When that baseline is established, status accounting must establish a record containing the date on which the baseline is established and the identification number, title, and issue date of the system specification. From then on, it must keep track of all SCNs/revisions and their applicable dates. The initial focus of status accounting on

the functional configuration identification is expanded as further baselines are established and as operational units are produced in a specific configuration and delivered to the using command. As with the functional baseline, information about each specification used to establish a baseline must be recorded and maintained. For the product configuration identification, information about the detailed documentation (e.g., drawings, or computer software listings) that define the exact design for each hardware and computer software configuration item must be recorded and maintained from the time they are initially released by the contractor, even though the Government does not take control of that identification until much later, when the product baseline is established. Identifying the technical documents and items is a part of the configuration identification process, but the actual recording and reporting of this information is a part of the CSA process.

In addition to maintaining this up-to-date record of, and historical data about, the technical documentation that describes the system and its configuration items, status accounting ensures that the data associated with configuration identification numbering is also gathered and recorded. The types of information to be recorded usually includes the configuration item identification number, configuration item nomenclature, part numbers, computer program identification numbers, national stock numbers, and other similar identification numbers for the system, configuration items, and component parts. Some of this information will be handled by the contractor, but much of it will be the responsibility of Government activities. Status accounting ensures that this information is available to support various processes (e.g., physical configuration audit, ECP review) in an understandable and useful format, and especially to support competitive spares ordering. This identification numbering data, as well as the

technical documentation describing the configuration identification, must be recorded, maintained, and updated (as required) so that current and historical information is always available as the system progresses through the acquisition life cycle.

Finally, CSA ensures that the information required to identify the exact configuration of delivered units is provided, and CSA maintains traceability of the unit's exact configuration while it is operated. The contractor provides the CSA information necessary to identify the configuration of all parts, to the lowest repairable level, installed in each delivered unit. Once the unit is delivered and placed in operational use the supporting and using commands are responsible for maintaining an information data base that tracks the exact configuration of each unit in operation, especially as the unit undergoes modification, maintenance, or retrofit activities. This information data base must be maintained until the system/unit is declared surplus, removed from the Government's inventory, and its raw materials recycled.

8.2.1.1 Status Accounting of Specifications. The program office should require the contractor to maintain a record of the most current approved status of all specifications that have been placed under Government control. In addition, during full-scale development the contractor should maintain the status of other important documents (such as the drawings and the elements of a CSCI's developmental configuration) which may not currently require official Government control but which are important to system development. The records being maintained for each specification should provide a history from the initial authentication of the specification, including a listing of all approved specification change notices and revisions made to the specification with the effective date of the change.

For each revision of a specification, the most readily available source for this information is the Specification Change Notice (SCN) form (DD Form 1696). An updated version of the SCN is inserted in the front of the affected specification when the change pages resulting from an approved engineering change are inserted into the specification (see paragraph 7.1.1.2.1). The SCN allows all holders of the specification to determine the most current approved status of that specification. The SCN lists information about the latest change to the specification, but it also provides a historical summary of previously approved changes to the specification.

For some programs, especially in the early acquisition phases, the SCN form may provide an adequate configuration identification data base. However, as the data base is expanded (and computerized) to incorporate the detail design documentation, that data base should include the specification information, too. The source of this data (usually the contractor) should be able to provide complete historical information on the changes (SCNs) to all revisions of all program specifications and to show the relationship between the engineering change proposals and the SCNs. It should list each revision of the specification, the approval dates for the revisions, and the dates for each approved ECP and SCN.

8.2.1.2 Status Accounting of Drawings. As with specifications, the contractor's drawings used to assemble the individual parts of the overall design undergo revisions during system development. Information about those drawings should be recorded and maintained beginning with the initial release, even though the Government may not control them for months or years. The data base should provide the most current approved status, as well as a historical record, of all of the different types of drawings required to manufacture the individual parts and the total assemblies. This information

will allow the system designers to determine the most current design from which they can address any developing changes to the design. In addition, this information would allow program participants to identify the drawings they should use, for example, in placing (or competing) a spare parts order or in inspecting a unit being presented for acceptance.

8.2.1.3 Status Accounting of Computer Software. One of the most difficult tasks associated with the development of a product is maintaining control of the computer software being developed. Because of its nature, a computer software program is easily changed. For this reason, the program office must insure that the contractor has an internal status accounting program in place that will establish and maintain a record of the current approved status of the software code/documentation, and a historical record of its development, for each computer software program developed and maintained for use in either the design of, testing of, servicing of, or manufacturing of elements/CIs of the overall system.

As with the other forms of technical documentation associated with a configuration item's identification, the record keeping with these computer software items should include current, and historical, information concerning the revision process, release date of each version/change, a cross-reference to any related engineering change proposal, and a listing of the technical documents' identification numbers, titles, and approval/authentication dates.

8.2.1.4 Status Accounting of the Overall System. For each CI produced and delivered to the Government, the information system should be able to provide a listing of the exact configuration of all parts that are installed/used in the CI. This listing does not

have to be down to the lowest-level individual components in the system design, but it should be able to identify the lowest-level repairable items/assemblies.

One approach used to provide this information is the hierarchical tree diagram for the overall CI/system (see Figure 8). (A similar depiction can be provided using an indentured listing of the documents for the overall CI/system.) At the top level blocks of the tree is the top-level CI/system identification, along with the corresponding current approved specification number/revision and perhaps the top-assembly drawing. The second level blocks of the tree identify the major subsystems (usually the major CIs or system segments) and the approved specifications which establish the allocated and product baselines for these subsystems and the top assembly drawing. The tree continues from level to level identifying the assemblies and components at that level and the related specifications and/or drawings for those components to completely define the top-level documents controlling each level of assembly. This information will provide an accurate record of the configuration currently being manufactured, accepted, and delivered.

8.2.1.5 Status Accounting of Operational Units. Once the product baselines have been established for the various CIs, and production units of these CIs have started being delivered, another very important aspect of CSA is begun. The information data base, in order to assist the supporting and using commands, must record and maintain information about the exact (part number) configuration of each delivered unit in the Government inventory. For each unit, the CSA process begins with an as-built/delivered record of the configuration of the unit. For the remainder of the

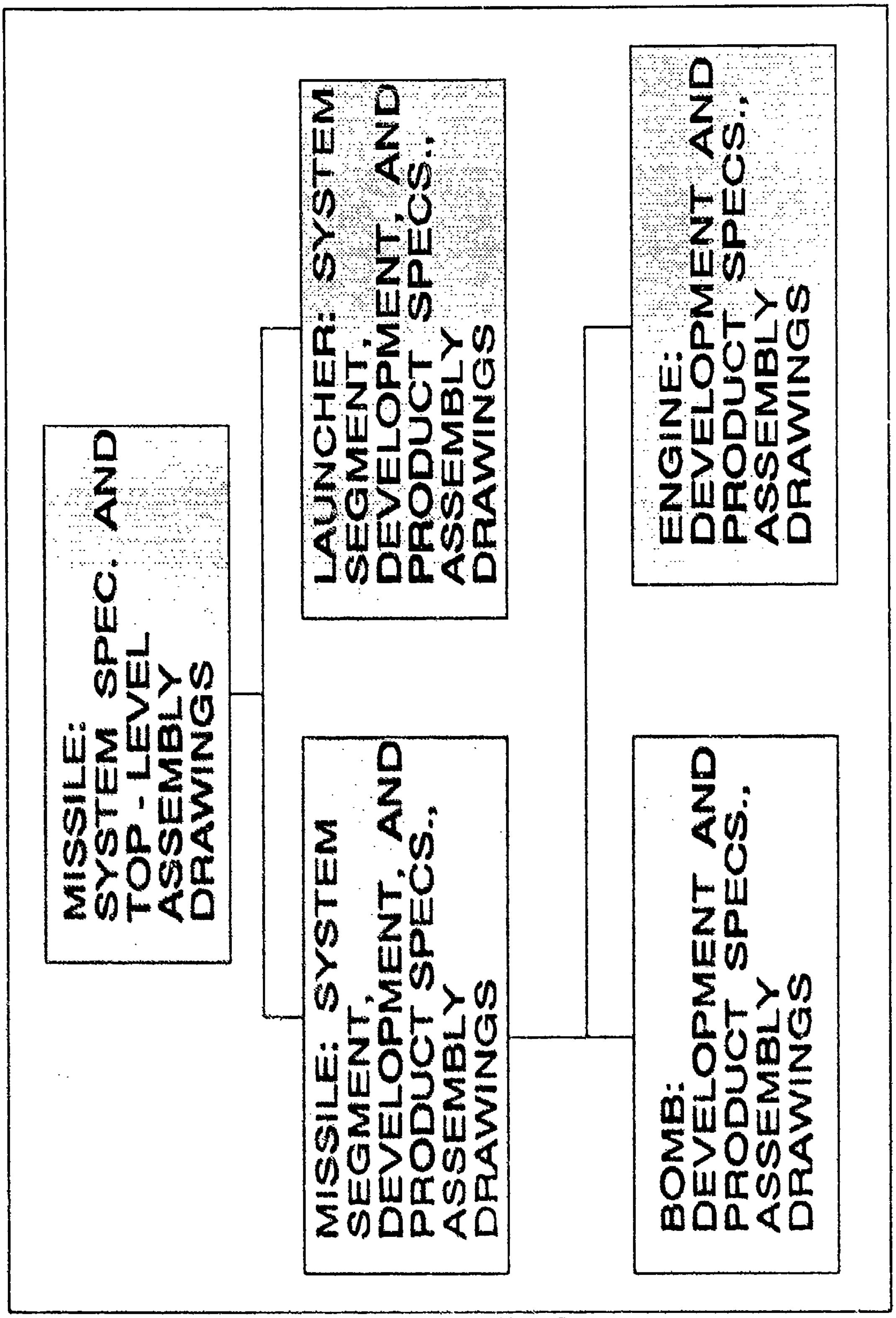

Figure 8: Hierarchical Tree Diagram

operational life of that unit, the data base must maintain that information for each individual operational unit as any factory or field retrofit is accomplished.

Additionally, the status accounting information should be able to record the removal and replacement of components during maintenance actions. In this role, the process must be able to track part numbers, serial numbers, and, for time sensitive components, the hours of operation for components installed/being installed in a configuration item or system. Configuration status accounting should be able to provide the supporting and using commands with complete information about the differences in the configurations of similar units in use, and about the number of units with each different configuration. Such information will enhance spares procurement and required modification/maintenance actions. If it doesn't provide this information, the logistics support of the system will be impaired due to the time lost determining the configuration of the unit, trying to locate/order the correct spares, and then locating the correct documentation to use for the modification (including retrofit) or maintenance action.

8.2.2 Relationship With Configuration Audits.

Configuration status accounting processes, as defined over the years in various regulations, manuals, and standards, have never formally included responsibilities for audit information. However, there is a considerable amount of information related to the preparation for, and the follow-up after, the audits which can be considered a part of the status accounting data base. The CSA data base can help to ensure that the configuration audits are scheduled appropriately and are accomplished in a timely manner. As program office and contractor personnel prepare for and conduct the

audit, the existing configuration identification information in the data base will be used to ensure that the design being audited is the most current approved configuration. The data base will also record the scheduled and actual dates for each audit. Any action items generated at the audits must be acted upon and closed out. The status accounting data base can be used to record brief information (e.g., identifying number and descriptive title) about the action item, track the action item's progress against interim milestones, and record the successful closeout of the action item as scheduled. Thus, the information data base will help the participants to remain aware of open problem areas and to focus management attention on those which are not being resolved in a timely manner. It will help to ensure that the audit is successfully completed and closed out expeditiously. [NOTE: A similar data base and action item tracking system would be needed by the engineers for the design reviews.]

8.2.3 Relationship With Change Management.

Recording and updating information about changes to the CIs and the program is the most dynamic part of the CSA data base. The data base must provide current, accurate information about the processing status of both technical changes (ECPs, deviations, waivers, ACSNs) and contractual changes (CCPs/TCPs, ACSNs) which occur during program development. As an aid to the process of configuration control of the baselined documentation, the status accounting data base must record and maintain information about all changes requested to the configuration item identification. The data base may also be required to record and maintain information about all change requests affecting contractual information (e.g., Statement of Work tasks, Contract Data Requirements List) about the program. Thus, CSA is responsible

for tracking any change to any of the stated program requirements, whether they are technical or contractual.

For both technical and contractual change management, CSA begins providing traceability of all change proposals from the time the change idea is first officially received and recorded, whether in the form of an ECP, CCP/TCP, ACSN, deviation, waiver, contractor letter, or any other agreed upon document. This traceability associated with change management is maintained and recorded until the change is disapproved or approved, officially incorporated into the contract documentation, and completely implemented as described in the proposal. The primary concerns of status accounting during system development are to expedite the review and processing of the change, to assure that the change is not lost while in review, and to reduce or eliminate any delays which might impact meeting the contractor's stated need date. Once the product baseline has been established and operational units have been delivered, the primary concern of status accounting is to monitor the incorporation of approved changes into the delivered units and the related logistics elements (e.g., spares, manuals, test software) to assure continued supportability of the new configuration in the field.

In each proposed change, the contractor provides a decision need date after which the proposal is no longer valid. This need date establishes the period of time during which the program office must review the change, make a decision, and officially direct the contractor to desist or proceed. Thus, the information data base for any proposed change must record the date of receipt of that change, the proposal number and title, the identification number(s) of the configuration item(s) affected, the affected contract(s), the stated priority of the change, and the stated need date. [Also see

paragraph 7.1.1.2.] The data base must reflect the date the change document and/or coordination letter was distributed to the members of the program office's CCB for review and comments, the scheduled date for the CCB (and any pre-CCBs), and the suspense date for the return of the coordination/comments. The data base provides a tool for managing the review of the change and provides an audit trail of the change during its review cycle in the program office. By recording the distribution of the change to the CCB members and the return of their comments and recommendations, the data base will allow the change manager to identify those organizations who have missed the suspense and who might need an additional reminder to complete their review.

Once the CCB decision is made, the data base should record the decision, the CCB Directive identification (with an identifying number and date), the date the CCBD is sent to the contracting officer, and the date that the official contractual authorization (or disapproval) is sent to the change's originator. If the proposed change was disapproved then upon the transmittal of the contractual letter explaining why the proposal was disapproved, the need for further tracking and updating of status information about that change ends, even though the information will be retained in the data base for historical purposes.

On the other hand, if the change proposal was approved, the data base must actively record and reflect the status of the change implementation activities. When the contractor submits the change proposal, it identifies all the impacts to the program documentation and provides a schedule for the accomplishment of the related activities. If the program is in the full-scale development phase, implementation may just involve distribution of the SCN. However, if the program is in the production phase

and the product baseline has been established and units have been placed in operational use, the implementation will be much more involved. Then, the data base must provide information to allow verification that the change is incorporated into the production line at the required date on the required unit. The data base must also record and provide information relating to updates to the various elements of the logistic support system. For example, this could include distribution of updated support software, delivery of new spares, and distribution of change pages for the technical manuals. If the change involves retrofit, the information would cover preparation of Time Compliance Technical Orders, actual delivery of the retrofit kits, and the rework of the units. Status accounting provides the means to track the accomplishment of the required actions to verify that they are performed as approved and scheduled, or to reflect/highlight any potential or actual problems for management action, in order to minimize any possible disruption of the system's operational/logistics capabilities.

8.3 Responsibilities of Participants.

8.3.1 Government.

The CSA responsibilities required of the Government are levied upon the implementing command, the supporting command, and the using command. The determination of which has primary responsibility, and of the information that each is required to provide depends on the phase of the acquisition life cycle.

8.3.1.1 Implementing Command. As directed by AFR 14-1, the implementing command (usually AFSC) is given the primary responsibility for CSA until Program Management Responsibility Transfer (PMRT) occurs. With the advice and concurrence of the supporting and using commands, the implementing command must select the

specific data elements, identify the data and the data analysis formats to be available, and the method(s) of record keeping. These must fill the needs of all the participants throughout the entire life of the system. These decisions are impacted by, and must consider the impact of, the number of CIs (hardware and computer software) selected for the product. Through the selection of these CIs, the program office influences the magnitude of the management effort which drives the size and the cost of the data base. In order to effectively select the data elements and data analysis capabilities, to be sure it will meet the needs of the CI/component managers throughout the life of the system, the program office must include the support activity (usually Air Force Logistics Command (AFLC)) in the process. This ensures that information required after PMRT is included, and prepared for, early in the generation of the status accounting data base capabilities. To minimize costs and data base proliferation, any existing contractor (or Government) systems that are already available and known to function correctly, should be utilized to the maximum extent possible. However, the resulting data base design and capabilities must ensure that the data base provided to AFLC is compatible with their management needs.

During the product's development, as the design evolves and expands from a concept to deployment and use as an operational system, the amount of CSA information that must be recorded and maintained increases. Initially, the Government may record and monitor the status accounting information by itself. As the detail design evolves, the amount of related data vastly increases. Then, the program office will often assign responsibility for recording and maintaining a significant part of data to the contractor. The CM personnel are required to review, and oversee, the contractor's data base to insure that the required information is available. However,

the Government activities will generate, record, and maintain a significant portion of the CSA data base with no contractor involvement (e.g., the tracking of the ECP review and approval process in the program office).

8.3.1.2 Supporting Command. The supporting command (usually AFLC) is normally given the primary responsibility for all CSA after PMRT. This responsibility is given to the Air Logistics Center (ALC) assigned the management/support responsibility for the system once it is in the operational inventory. AFLC has established partial CSA procedures and data bases to record and report maintenance and retrofit/modification actions as a part of the logistics support of the system. However, the existing CSA capabilities must be supplemented to provide continued availability of a comprehensive CSA data base throughout the operational life of the system, as required by AFR 14-1.

Depending upon the system being developed (e.g., aircraft, multi-application commodity, engine), AFLC has already developed mechanized CSA systems, designed with varying capabilities, that account for the current exact configuration of delivered products and/or equipment. By bringing the AFLC System Manager on board early to help in the selection of CSA data elements and analysis capabilities, the program office can avoid duplication of effort which would result if data available from AFLC were ordered from the contractor. AFLC must advise the program office about typical data, records, and reports it maintains once the system is placed in operation. The program office should use these inputs to ensure that the information ordered from the contractor will provide the necessary current and historical information needed to supplement the required status accounting information maintained by the supporting command.

As the program proceeds through the production/deployment subphase and enters the operational support subphase of the system acquisition life cycle, the supporting command must eventually assume the CSA responsibilities for maintaining the data base. Initially in the production phase, they will have to provide their normal elements of the information data base while also reviewing, and providing feedback to the program office, about the information in the contractor's data base. For as long as there are production contracts in effect and production deliveries being made, the contractor is usually required to provide current and historical data relating to configuration documentation, to item identification numbers, and to the status of approved change proposal implementation. For each delivered unit, the contractor must provide an accurate listing of the exact delivered configuration. The supporting command (or the using command) must input and maintain the information about the current configuration, and location, of each operational unit and the status of maintenance (and other logistics support actions) against those units. But as the production phase concludes and deliveries are completed, the contractor is no longer required to input and maintain this data. The supporting and using commands must pick up the responsibility for maintaining the data formerly handled by the contractor as well as continuing to input and maintain the information about the current configuration, and location, of each operational unit and the status of maintenance (and other logistics support actions) against those units.

8.3.2 Contractor.

During system development and production, the prime contractor (or integrating contractor, if one has been assigned) is responsible for recording and maintaining the contractually-required information and for making the information available via the

contractually required medium. During the time that the contractor(s) are involved with the program, they will typically be required to record and maintain information that (1) provides traceability of the specifications and other configuration documentation, (2) tracks all internal design documents (even those that are not (or not yet) formally placed under Government control), (3) provides traceability of item identification numbers; (4) maintains a record of all approved change proposals (separated into those that are technically oriented and those that are contractual in nature), (5) tracks the implementation of all approved changes to ensure that the required actions are accomplished in a timely manner, and (6) provides and maintains an accurate listing of the exact configuration of each test prototype and each deliverable production unit. The contractor should provide status accounting information in a format that is understandable and usable and that can be integrated with other information systems.

8.4 <u>Depth Of Status Accounting Information Recorded.</u>

The accuracy and currency of the CSA information, and the availability of related management capabilities, are major factors in the effectiveness of a program office's developed configuration management system. Initially, status accounting will record the establishment of configuration baselines (beginning with the functional baseline) and maintain information about the baselined documents. As the program proceeds through its development, status accounting must record information about the status of the processing of proposed changes to each CI/CSCI and about the implementation status of approved changes. Prior to the establishment of a product baseline, the status accounting information is focused almost exclusively on documentation describing, and activities involving, the individual CI/CSCIs. After the product baseline has been established, the status accounting information will continue to address the

CI/CSCIs, but it will focus more on recording and maintaining information about individual units of the CI/CSCIs and about their constituent parts.

8.4.1 Prior to the Product Baseline.

During the early stages of systems design, status accounting information must include a record of the progression of development of each CI/CSCI. The data base should provide (a) the current status of each configuration item's development and of its specifications and/or other technical documents, and (b) the status of program actions (e.g., approved changes) to the documents that are used to identify the configuration of each of the hardware and computer software items.

8.4.2 After Product Baseline.

Once the program office establishes the product baseline and the product configuration identification, the CSA records and reports begin addressing the configuration of the entire system. The program office, usually be levying the requirement on the contractor, is responsible for maintaining the status accounting records for (a) the configuration identification for the items that comprise the total system, (b) the approved change proposals that have been authorized for incorporation into the CIs, and (c) tracking the status of the implementation of all approved changes listed in (b), such as the status of TCTO preparation, of the release of revised drawings, of the availability of new spares, of updated pages for the manuals, and of revised automatic test software. After PMRT has occurred, the responsibility of making sure this information is available is passed on to the supporting agency, although the contractor will usually continue to be the source of the information until the production deliveries are completed.

Again the exact format of the documents that report this information is left for the program office and contractor to produce. Yet, it should be noted that once the product baseline has been established, the CSA process will normally utilize a computerized management information system data base. This data base should be detailed enough to allow the implementing command program manager, or support command system manager, to produce various forms of status reports (e.g., retrofit kit availability status, active changes status, specification and drawing revision status, spares purchasing and distribution, and modification accomplishment).

9. APPLYING CM FOR FULL-SCALE DEVELOPMENT

Sections 4 through 8 have described the role of configuration management in the development of a product as it progresses through the system acquisition life cycle, and they discussed the four processes that comprise this combined technical and managerial discipline. The following paragraphs will discuss those actions, and responsibilities, that a configuration manager can perform for the program office during full-scale development to provide a successful configuration management program. To provide a successful CM program, the configuration managers must require that some of the requirements associated with the principles of the four CM processes be performed or monitored for the program manager by the program office's CM personnel, some will be performed/monitored by other program office functional personnel, and many of the CM requirements will be performed by the contractor.

This section will first address those responsibilities that CM personnel can perform within the program office to help the program manager successfully acquisition the required weapon system. This will include outlining the requirements that are included in program management plans and providing outlines of specific Operating Instructions that can be used in the program office to describe what configuration management personnel (as well as other program office functional personnel and program participants) should be doing to accomplish the CM portion of system development. Afterwards, suggestions will be provided as to what types of CM requirements should be levied upon the contractor through the use of Statement of Work tasking paragraphs. [NOTE: Throughout this section, there will be places where (1) suggestions will be provided to the configuration managers to help them decide on which alternatives to take, and (2) the configuration managers need only fill in specific

program information for their program office. These places are set aside through the use of angle brackets (< >). In addition, various amplifying comments, intended to guide the generation of the wording rather than providing a wording example, are set aside in square brackets ([ ]).]

9.1 CM Within the Program Office.

If the program office's configuration management process is developed, and maintained, correctly, it can be the most beneficial technical management control tool at the disposal of the program manager. As the program enters, and proceeds through, full-scale development, configuration management should provide the program manager with (1) the identification of the functional baseline and its associated technical documentation; (2) the means to verify and validate that the requirements are appropriately allocated from the system to its configuration items prior to establishment of the allocated baselines; (3) the assurance that the baselined technical documents that define the system and the CIs are maintained and controlled, unchanged, unless the Government approves the changes; (4) the traceability of the system requirements from the functional baseline, through the allocated baseline, to the product baseline; and (5) the traceability of any program change actions taken against these baselines. To provide these functions for the program office as it enters, and accomplishes, the full-scale development effort, configuration managers should provide inputs to program management plans that will describe the CM activities/support required of the various participants on the program (including those for the functional activities within the program office and those for other commands/agencies); and they should also generate Operating Instructions (OIs) for the program (and for other program

participants) that describe the responsibilities that configuration management (through actions by the appropriate personnel) will perform for the program.

9.1.1 Program Management Plan (PMP).

The PMP is the principal document by which the program manager defines and obtains agreements about the management responsibilities of the functional offices and the participating commands/agencies involved in the program. It is developed (using inputs from program office functionals and other program participants), approved, and issued by the program manager to show the integrated, time-phased tasks and resources required to accomplish the tasks specified in the Program Management Directive (PMD) and the command supplements to the PMD. [According to AFR 800-2 (dated 16 September 1985), a PMP is not required for programs that include basic research, exploratory research, or advanced technology programs in laboratory environments.] When completed and signed, it is furnished to higher authorities by the program manager to provide them information on the program.

The PMP must clearly and explicitly state the program objectives, schedules, tasks, risks, participants and their interrelationships, resources required, and an overall strategy. It is developed as a unified plan to help the program office, supporting command, and operating command personnel work toward a common goal. It should define the support required of all participating organizations.

9.1.1.1 Configuration Management's Portion. The configuration manager's portion of the PMP establishes the program office's internal configuration management plan for establishing and maintaining a viable CM program during the development of the weapon system. It should outline all of the participating agencies (especially the ones

responsible for accomplishing or supporting portions of the CM function) and their roles. Usually, the configuration management input is a sub-part of Section 4 (which also includes system engineering).

[NOTE: The sections providing this information should be written in relatively general terms such that the PMP will not require revision every time the program office decides to make a minor change to the CM practices. The CM organization should use the OIs as the means to provide the specific guidance for the program for each area covered in the PMP.] For full-scale development, the CM portion should describe: (1) how, and when, the CM functions will be implemented and accomplished; (2) what tools CM personnel will use to apply the principles of configuration management; (3) the CM directorate's organization [to include the program office Configuration (Change) Control Board and/or Software Configuration Control Sub-board] in terms of division of personnel resources to accomplish the different CM functions; (4) the current, and future, specifications required for the program; (5) what other documentation will be baselined for change management; and (6) if any interface control requirements exist for the program, what the program office's role (with respect to configuration management) should/will be in this interface environment.

9.1.1.2 Suggested Inputs for Full-Scale Development. The following paragraphs contain suggested information that a configuration manager, or configuration management directorate, might want to include in the PMP for a program entering full-scale development. [NOTE: The wording provided can be used almost "as written" in the following paragraphs when preparing the CM portion of the PMP. The use of these, and other, paragraphs should be decided upon based on the acquisition strategy being sought by the program manager for each particular program.]

a. Overview - Configuration management is responsible for identifying (documenting) the configuration of the <weapon> system and its configuration items (CIs); conducting configuration audits to verify and validate the configuration of each CI; controlling any changes to the <weapon> system and/or CIs through the Configuration (Change) Control Board; and maintaining the status of all engineering changes [to include engineering change proposals (ECPs), deviations, waivers, and advanced change study notices (ACSNs)] through some form of management information system.

b. Organization - The <weapon> Directorate for Configuration Management has the overall responsibility for the identification, audit, change management, and status reporting necessary to manage the hardware and software configuration items (CI/CSCIs) and associated technical documentation for the <weapon> system, its CIs, and any support equipment.

The Directorate of Configuration Management may be aligned into separate divisions/branches to accomplish the tasks associated with the requirements of each CM process. For most programs entering full-scale development, it seems that the change management/change proposal processing responsibilities of CM are such that this process should be broken out into its own division/branch. In addition, as the program progresses through full-scale development, the audit function will begin to increase in importance. It would be beneficial to also separate this CM function into its own division/branch to take care of the audit tasks for each of the developing CI/CSCIs. The other requirements (configuration identification and configuration status accounting) are such that the tasks associated with their portions of the CM process

could be performed throughout full-scale development by personnel in a consolidated CM division/branch.

c. Procedures - To accomplish its tasks, the <CM organization> will implement the current procedures outlined in [NOTE: find the latest/most current version of each] AFR 14-1, AFR 800-14, MIL-STD-480 (or MIL-STD-481, if applicable), MIL-STD-482 [NOTE: This document may be rescinded in the future. As it is now, many programs have followed its basic information but have allowed the contractors to deviate and provide their own formats, as long as they are able to interrelate these formats with other Government information systems.], MIL-STD-483, MIL-STD-490, MIL-STD-1521, and, if required, DOD-STD-2167. In addition, the <CM organization> will also ensure the procedures outlined in the contractor's Configuration Management Plan <if one is required/prepared for the program> are implemented.

d. Management principles:

1. Configuration identification: The <CM organization> will manage the specifications, drawings, and other technical documents that identify the system and its CIs. The functional baseline of the <weapon> system has been established <assuming that the functional baseline has been established and the corresponding functional configuration identification noted> using the <weapon> system specification. The allocated baselines of the CIs and support equipment will be established using the appropriate development and product specifications (either as separate documents or as Two-Part Specifications). The <engineering function> will assist in the review of the contractor's draft specifications.

The development specifications for hardware CIs will be approved at the System Design Review(s) <or not later than the CIs' Preliminary Design Review(s)>. <If the

program includes computer software CIs (CSCIs)> The Requirements (and, if required, Interface) Specifications for the CSCIs will be approved at the Software Specification Review (if one is conducted) but no later than the Preliminary Design Review for each CSCI. The product specifications will be approved at the Physical Configuration Audit(s). [NOTE: The PCAs are often accomplished for computer software during full-scale development and for hardware during production, so procedures for software PCAs should be included in the PMP. If the Physical Configuration Audit is to be a production phase activity, this type of statement may be omitted until the PMP is updated for the production/deployment phase.]

2. Design reviews and configuration audits: The <CM organization> will assist the program manager in establishing contractual configuration baselines at the appropriate program design reviews; validating the development of the CIs (and the system) at the Functional Configuration Audit(s) <and, if needed, at a Functional System Audit>; and establishing product baselines at the Physical Configuration Audit(s). The configuration managers will conduct the audits and assist/monitor the systems engineers for the design reviews.

3. Change (configuration and change control) management: The <CM organization> will establish change control procedures (in accordance with AFR 14-1 and MIL-STD-480 or MIL-STD-481) to manage changes (engineering changes) to formally established technical baselines and changes (task/contract changes) to other contractual (non-technical) documents. The functional baseline is defined by the <weapon> System Specification(s). The allocated baseline(s) will be defined by the corresponding development (or requirements for computer software CIs) specification(s). [NOTE: If the acquisition strategy being pursued is to conduct

Physical Configuration Audit(s) during full-scale development, then include a statement that also states that the product baseline(s) will be defined by the appropriate product specification(s).] Changes to these baselines, (and changes to the contract) will be controlled through the approval of an Engineering Change Proposal (or a Contract/Task Change Proposal) by the <weapon> Program Office Configuration (Change) Control Board (CCB). The <CM organization> will provide technical and administrative information/assistance to the CCB Chairperson.

[NOTE: Another aspect of change management that may have to be discussed is the possibility of allowing the establishment of a lower-level CI's allocated baseline to be delayed until the CI's Functional Configuration Audit is performed. If this type of development strategy is pursued by the program office, it allows the contractor flexibility in developing, and altering the requirements in the tentatively approved specifications for those lower-level CIs. However, even if this approach is acceptable, those higher-level CI specifications that incorporate the requirements being passed on to these lower-level CIs must be baselined and controlled to provide some managerial/programmatic control of the overall system/CI design.]

4. Configuration status accounting: The <CM organization> will ensure that the program's configuration status accounting information system provides, at a minimum, the configuration of each CI and its related documentation from the point in time when it was created or baselined through its current state. The information system will also identify proposed change actions to the system and individual CIs; the status of proposed actions on the changes; the status of approved changes not yet completely implemented; and highlight any delinquent actions in implementing

approved changes with the reasons for the delinquency. The information system will provide data about the current configuration of each test unit.

e. Interface control (if required) - The overall interface requirements for the <weapon system> are contained in the <weapon> System Specification, the Prime Item Development Specifications, and the Software and Interface Requirements Specifications. They define the contractually binding functional and physical interface requirements for the program. Any interface control between associate contractors, below the level of the specified interfaces, will be managed through Interface Control Drawings/Documents (ICDs) and an Interface Control Working Group (ICWG). [NOTE: If the interfaces are of a very complex nature, the program office may consider incorporating more of the interface requirements/documents in the established baselines under Government control. If this is the case, there will be fewer non-contractual ICDs under the control of the associate contractor members of the ICWG.]

f. Plans - The <CM organization> will maintain control of the contractor(s) configuration management program by requiring the submittal of a Configuration Management Plan for approval. The contractor will be required to describe in detail the internal organization and procedures that will be used to implement CM throughout the program. This plan will also be required to outline the working relationships between the program office and the contractor to ensure that all CM activities are effectively conducted. However, the plan normally will not be incorporated into the contract as a set of contractually binding tasks. It should be included as a contractual deliverable with a corresponding Contract Data Requirements List enclosure to the Statement of Work tasking paragraph.

9.1.2 Computer Resources Lifecycle Management Plan (CRLCMP).

In addition to the PMP, the configuration manager needs to investigate the possibility that the program is using a CRLCMP. The CRLCMP (which is a Government-developed plan) documents the development strategy being undertaken by the program office for the program's computer resources and especially for the future support of the software. If there is a CRLCMP, if it plans the use of a Software Configuration (Change) Control Sub-board for computer software engineering changes that do not affect any other hardware CIs, then the configuration manager must ensure that this practice is included in the configuration control portion of the PMP (along with any other restrictions for computer software CIs). The CRLCMP must be approved prior to the release of the full-scale development request-for-proposal.

9.1.3 Operating Instructions.

The PMP includes the program office's internal CM program, described in general terms, that will be conducted by the assigned CM personnel. To provide specific direction on the tasks and policies to CM personnel, as well as to other members of the program office and/or other program participants, the CM organization will normally prepare Operating Instructions (OIs) and submit them to the program manager for approval and issue. These OIs will document the detailed application of the CM processes for the product (program) under development. The following paragraphs discuss typical OIs that should be prepared for a program entering full-scale development by the CM organization for approval by the program manager. [In some programs, the following OIs may have been initially prepared during the concept demonstration/validation phase of the system acquisition life cycle. If this is the case,

then the OIs need to be reviewed for their adequacy as the program enters full-scale development. If not, then the OIs may have to be prepared as discussed below.]

9.1.3.1 Specification Management OI. The principle of configuration identification for the program office dictates that the CM organization manage the program's specifications. This OI should indicate that, with technical assistance from other program participants, the program office's CM organization will administer the review, processing, coordinating, baselining, and controlling of all specifications submitted to the program office. For most major programs under development, the program office will have authenticated the System Specification during the concept demonstration/validation phase of the system acquisition life cycle. In any case, the specification management OI prepared should include provisions to control and update those specifications that are already approved and authenticated, as well as the ones that will be authenticated during this phase. The specification management OI should reference the Change Management OI to address requirements for processing changes to those specifications that are already approved and authenticated.

Most full-scale development programs have prepared some form of this OI for the program office. The following paragraphs provide suggested sections that can be included in a specification management OI. These paragraphs need not be used by every program office, but they should be reviewed as suggested inputs for an OI. The specific content of this type of OI will differ for each program depending upon the acquisition strategy being pursued on, and the current state of development of, the system under design.

a. Overview - Provide an overview that states that the intent of the OI is to prescribe the policies, procedures, and responsibilities of the <weapon system>

program office organizations for processing, reviewing, and controlling contractor prepared specifications applicable to the <weapon system>.

b. References - If there are any other program office OIs that may interrelate with this OI, list them here. Some examples would be a Configuration (Change) Control Board OI, or a Change Management (Change Proposal Processing) OI. Also, list those regulations, standards, and pamphlets which this OI is supplementing.

c. Definitions - A separate section can be included to provide definitions of the different types/forms of specifications that may be encountered by personnel associated with the program.

d. General Policy - Provide a paragraph that defines the basic interrelationships among the different program office functions concerning specification management. This usually includes a statement of the requirement that the <weapon system> Directorate of Configuration Management is responsible for identifying and controlling the specification type, format, and maintenance. In addition, the authority for recommending approval for authentication is normally vested in the CM organization as well. However, the technical content (to include qualification and acceptance test requirements) should be approved by the engineering function within the guidelines specified in this OI. In some cases, when conflicts arise concerning the approval/ disapproval of a submitted specification, there may be a need to include a statement concerning the use of a Specification Review Board (similar to the CCB, if not the same body) to resolve these conflicts. Once specifications have been submitted and authenticated, require that any changes to these specifications caused by some related Engineering Change Proposal (ECP) will be acted upon concurrently with the ECP in accordance with the policy listed in the Change Proposal Processing OI.

e. Specific Policy - [This section should discuss the specific approach (approaches) being pursued by the program office in establishing specifications. Some issues that the configuration manager should consider addressing (including statements describing the specification activities associated with these issues) are]:

1. Prior to developing subsystem or end item CI/CSCI development/ requirements specifications, the contractor for the program will be required to submit a specification tree (hierarchical form) that describes the recommended specification approach. [NOTE: In most instances, the contractor will have submitted this tree during the concept demonstration/validation phase of the system acquisition life cycle when the program went through its System Design Reviews.] This tree should identify the specification number, the related CI or CSCI, and the type and form of specifications to be used to document the requirements. Upon receipt of this specification tree, the CM organization should convene a meeting with the program manager and all affected functionals to reach an agreement on the specification approach. After the program manager makes the final determination of the specification approach that will be followed, the approved specification tree should be included in the appropriate required deliverable document (normally, the Configuration Management Plan and possibly the System Specification).

2. State what specifications will be prepared, processed, and maintained for the newly identified system segments, CIs, and/or CSCIs in accordance with the provisions of MIL-STD-490 (with any tailoring provided with the contract data requirements list, CDRL). If required, discuss the approval process required for contractor equivalent formats.

3. A statement should be included as to the requirements for maintaining documentation related to any previously developed systems, subsystems, CI/CSCIs, Government furnished equipment, or end items being used. Generally, if a previously developed component requires modification before it can be used by the program, the process for developing addendums or appendices to previously approved specifications should be discussed. [Try to require the development of new specifications only when there are substantial differences in design criteria and performance requirements between the end items. The CM organization, based on recommendations from the affected functionals, should make the determination.]

4. State how the specifications being developed for the program will be controlled. The approach normally followed is for contractor internal control prior to authentication, and for MIL-STD-480 ECP control after authentication.

5. State who has signature authority for specification authentication. This authority is generally given to the program office's chief engineer, but it must be delegated officially by the procuring contracting officer.

6. Discuss the time frame when the different specifications should be submitted, approved, and authenticated. By when are the development and requirements (and interface, if required) specifications required to be authenticated <in most instances for the top-level CIs, this should be accomplished after the SDR and no later than the CI's Preliminary Design Review. For those lower-level CIs, these specifications may be authenticated at the same time or they may be tentatively approved no later than PDR and authenticated at the CI's Functional Configuration Audit>. Since draft product specifications are required on the program during full-scale development, discuss the required activities <normally submittal and review of drafts>.

f. Procedures/Responsibilities - The OI should also discuss the procedures and responsibilities required of each of the functionals within the program office for specification management. These include:

1. Directorate of Configuration Management: Some of their responsibilities are: (a) to manage and direct the review and authentication of draft specifications with the coordination of other appropriate functional areas; (b) verify that the format and content of the specifications are in accordance with MIL-STD-490; (c) make sure that a technical coordinator is designated to incorporate all agreed upon comments during the review of the draft specification <this individual should also be the one used later when changes are submitted against the specification, as discussed in the change management operating instruction>; (d) ensure specifications are presented to a Specification Review Board (or CCB), if required; (e) document the actions of the board and record these results on a Configuration Control Board Directive (AFSC Form 318) <a change control task that is usually also included in a change management operating instruction>; (f) prepare contract's letter that transmits the authenticated specification and directs the contractor to distribute it <in accordance with the CDRL> and to add the specification to the specification list; (g) maintain specifications current with any approved changes; (h) maintain status accounting information of all specifications and changes to these specifications; and, if requested, (i) issue status reports.

2. Directorate of Engineering: This directorate is normally required to: (a) designate a technical coordinator for each specification who is responsible for integrating comments from all respondents into a consolidated set of requirements/changes; (b) coordinate with the CM organization to maintain all schedule

requirements for review of the proposed specification; and (c) authenticate the specification for contractual incorporation.

3. Directorate of Manufacturing and Quality Assurance: This directorate is normally required to: (a) verify Section 4 quality assurance provisions <especially as they relate to acceptance tests, classifications of defects, and sampling> are sufficient; (b) review for appropriate packaging and handling requirements; (c) provide comments to the technical coordinator; and (d) coordinate on the final specification authentication correspondence.

4. Directorate of Integrated Logistics Support: This directorate should: (a) review the specification to ensure logistics requirements, listed in paragraph 3.5 (and logistics-related requirements, including reliability and maintainability), are acceptable; (b) obtain any AFLC or Air Logistics Center comments required; (c) provide comments to the technical coordinator; and (d) coordinate on the final specification authentication correspondence.

5. Program Manager/Project Manager: This person should: (a) review the specification for compliance with program requirements; and (b) coordinate on all specification correspondence to the contractor.

6. Directorate of Contracting: This directorate should usually: (a) be responsible for contractual wording requirements; and (b) signing and releasing/issuing all direction to the contractor relating to the contractual status of the specification.

7. All remaining directorates within the program office are responsible for soliciting remarks and comments from their supporting agencies, combining their comments with those of their supporting agencies, and providing these comments to

the technical coordinator. They should also coordinate on all correspondence being sent to the contractor.

9.1.3.2 Configuration Audits OI. Another way in which the CM organization can help the program manager during full-scale development is through the use of the configuration audit process. The CM organization assists the program manager to: (1) validate the development of the CI/CSCIs (validate that their actual developed performance complies with the development/requirements specifications) by conducting a Functional Configuration Audit (FCA) for each CI/CSCI and, if needed, by conducting Functional System Audits (FSAs) <in those cases where system level audits are required to verify complex system-level performance requirements have been met>; and (2) verify that the "as-built" configuration of each CI/CSCI <by ensuring that the CI/CSCI reflects the design cited in its product specification before establishing its product baseline> by conducting a Physical Configuration Audit (PCA) for each CI/CSCI.

In the past, most programs in full-scale development conducted FCAs on CI/CSCIs sometime during that phase of the system acquisition life cycle. On most major programs, where the contractor performing system development has been awarded system production, the FCAs and FSAs have not always been completed before at least limited production is authorized. Some programs will require that partial FCAs be accomplished for key elements (CIs) of the system before such limited production is authorized. Most programs will conduct the final FCAs (and PCAs) first on the lower-level CIs before conducting them for the higher-level CIs. For the FCAs, this permits the results (and action items) from the lower-level audits to focus the correction of the problems on their source CI. Similarly for PCAs, audits of the lower-level CI detail

design documents reduces the need for complete disassembly at the top-level PCA. For this reason, the following suggested paragraphs will primarily address the issue of FCA during full-scale development, but they will also include possible PCA actions that the CM organization may want to include <depending upon the acquisition strategy being pursued by the program office>.

a. Overview - Provide an overview that states that the intent of the OI is to establish policies, procedures, and responsibilities for the FCAs and, if needed, FSAs; <and, if applicable to this phase PCAs for each CI/CSCI>. [NOTE: Since this is a program OI, it may be well to establish the procedures for all three audits, even though the PCAs usually won't be conducted until the production phase. If adjustments to the PCA procedures are required later in the program, the OI can be changed later and reissued.] The stated policies, procedures, and responsibilities should ensure that the audits will be accomplished in an efficient and effective manner. If appropriate, the OI may include requirements for support from additional contractors (e.g., independent verification and validation), participating activities, and other systems which, although not directly responsible for the actual conduct of the audit, will assist in the accomplishment of the audit. This section should list any regulations, standards, or pamphlets that this OI is supplementing.

b. Definitions - A separate section can be included to provide the program office definitions of what each audit will be used for.

c. Policy - This section should describe the program's policies regarding the accomplishment of each CI/CSCI audit. It should also state who will chair the audit and the responsibilities of the participants. [In most instances, this is a statement that the Directorate of Configuration Management will chair and conduct each audit for the

<weapon system> program office. However, in the case of an audit of a subcontractor's CI, the contractor will often chair the audit (for the Government), and the Government participatory roles will have to be defined.] Some other issues that are normally addressed in this area are:

1. State the intent of the FCA. [Usually, this is a determination that the item's performance requirements have been met and that the item has been successfully qualified.]

2. If PCAs are included, state the overall policy for the PCA. [Usually, this is a determination that the assembled configuration of the unit of the CI matches its product baseline documentation.]

3. Summarize the requirements and audit procedures described in MIL-STD-1521 which are being adhered to and cite the specific tailoring that applies to the audits.

4. Include an audit checklist that will be used at each CI/CSCI audit. <NOTE: Some sample checklists are provided in the appendices of MIL-STD-1521.>

5. Include or reference the certification materials to be used <normally it is a signed certificate or set of such certificates similar to Figures 3 and 4 in MIL-STD-1521> to certify completion of the audit. Identify the functions/organizations delegated the responsibility for certifying completion of various parts of the audit. Identify the organization for certifying completion of an audit for a subcontractor CI.

6. Include a statement that requirements for the generation of unique data should be kept at a minimum. [It is normally a policy to minimize additional specialized data, or contractor effort, over and above the requirements of MIL-STD-1521 for the sole purpose of the audit.]

7. Discuss the program office plan for combining audits for smaller CIs or for incremental audits for large/complex CIs. Will there be one for each CI/CSCI? Will the program office try to combine some of the audits together?

8. State the requirement for maintaining information on the status of CI/CSCI specifications prior to the start of an audit. If the plan is to delay authentication of the specifications for lower-level CIs, do the development specifications have to be authenticated and under Government control prior to FCA?

d. Procedures - What procedures will be followed for the conduct of the audits?

1. Is the contractor required to forward planning data identifying CIs (or groups of CIs) to be audited; their time phasing, including the recommended date(s) for the audit(s); and the specific actions to be performed? How many days prior to the start of the audit should the planning cycle be started? According to the CDRL, when should the final agenda be distributed to the program participants?

2. What tasks will the CM organization perform prior to actually convening the audit(s) on the agreed-to date(s)? Will a pre-audit meeting be conducted (prior to or at the start of the audit) with the affected program participants to discuss: (a) orders, travel arrangements, and hotel accommodations; (b) the planned conduct of the audit; (c) the responsibilities of each participant (especially of team leaders and members of the Government Executive Panel); (d) any known or suspected (potential) problem areas; and (e) the understanding that any changes to an established Government position concerning the audit will not be discussed in front of the contractor, but will first be addressed in a "Government-only meeting."

3. Conduct of the audit. How will the audit begin? Will there be an in-briefing by contractor to discuss the audit agenda and proceedings? After an in-briefing, if

used, the audit process usually followed is: (a) contractor chai, rson should introduce the contractor's team and give the latest information pertaining to the CI; (b) program office chairperson of the audit team (usually a CM organization individual) should introduce the Government team, and review the purpose, scope, and objectives of the audit; (c) a combined contractor-Government meeting to logically divide both teams into groups to accomplish certain audit tasks; (d) begin the audit, each team should record discrepancies on worksheets and submit them to the Government audit chairperson; (e) Government Executive Panel will determine the applicability of the discrepancy and reject it or submit it to the contractor; (g) contractor provides response to the discrepancy; (h) Government evaluates the contractor response; (i) contractor and Government discuss and agree on the status of the discrepancy and any residual actions (action items) required; (j) the team leaders and/or audit chairpersons should sign the certificate(s); and (k) chairpersons should review and sign the minutes of the audit, making sure the minutes contain a "Disclaimer" stating that only a contractual letter from the procuring contracting officer will establish any baselines (for PCA) or provide any changes in contractual direction.

4. What post-audit actions will be performed? Is the contractor required, by a Contract Data Requirements List (CDRL) tasking, to distribute minutes of the audit? If so, what should be found along with the basic minutes (e.g., agenda, list of participants, copies of presentations, team worksheets, certificates).

5. Who is designated to monitor activity on/closeout of the action items?

e. Responsibilities - The OI should also discuss the responsibilities required of each of the functionals within the program office for the conduct of the audits. These include:

1. Directorate of Configuration Management: Some of this directorate's responsibilities include: (a) providing or delegating an audit chairperson; (b) monitoring overall contractor support and specific management/approval of all formal configuration audits; (c) identifying the CI(s) to be audited; (d) providing information about the approved configuration identification which is to be the basis for each audit; (e) organize Government team and assign their responsibilities; (f) notify contractor of final approved audit date; and (g) for PCA, ensure that CI/CSCI product specification(s) have been received, and reviewed, prior to the start of the audit.

2. Program/Project Manager: This person is ultimately responsible for the overall conduct of all assigned projects within the program. The program manager should: (a) ensure that provisions are incorporated into the contract for the conduct of audits; (b) ensure that the contractor has provided required resources for the audits; (c) ensure configuration audit milestones are included in contractual schedules and are being tracked; and (d) coordinate on formal notification to contractor of audit accomplishment.

3. Directorate of Engineering: This directorate is normally required to: (a) provide technical support to the CM organization for the audits; (b) review specifications, drawings, computer code, test procedures, test reports, qualification/acceptance testing data, and other documents to ensure that the CI/CSCI has met its specified requirements; (c) for PCAs, ensure that "as-built" configuration

matches the "as-documented" configuration and that any differences are documented; and (d) coordinating on all decisions and contractual letters.

4. Directorate of Manufacturing and Quality Assurance: This directorate will provide support primarily for the PCA; it often works in concert with the plant representative personnel to support the PCA. If PCA procedures are being included in the OI, it should be included. Some of its responsibilities include: (a) inspecting the DD Form 250 for accuracy; (b) reviewing contractor's manufacturing control and engineering release system; (c) identifying producibility improvements in support of audits; (d) assisting in verification that the "as-built" configuration of the CI/CSCIs conform to their documents; and (e) coordinating on all decisions and contractual letters.

5. Directorate of Integrated Logistics Support: This directorate responsibilities include: (a) reviewing the test results and providing inputs as to whether the actual logistics capabilities of the CI meet the logistics requirements/constraints listed in the development/requirements specifications; and (b) coordinating on all decisions and contractual letters.

6. Directorate of Contracting: This directorate would usually have no direct responsibilities involving the audits, but they should usually be responsible for signing and issuing all direction (to include establishment of the product baseline after each PCA) to the contractor.

7. All remaining directorates with the program office should be responsible for providing comments, and coordinating on, all contractual letters.

9.1.3.3 Program Review Boards. As the system/CI undergoes development, the program manager may require that different types of program-related decisions be

made. Decisions must be made regarding proposed changes to currently approved technical (those that identify established baselines) and/or contractual (those that establish required taskings) documents. Other decisions involve the review of the results of work accomplished by the contractor, in response to action items established at a review or audit, in order to determine whether the contractor has successfully completed that action.

To assist in making decisions about changes to current, approved technical documentation, the program manager will usually require the expertise provided by the program office's Configuration (Change) Control Board (CCB). Every program office managing the development of a weapon system must have a CCB that is responsible for regulating the change management process and for making change decisions (see paragraph 9.1.3.3.1). Configuration (Change) Control Board activities may also include inputs from a second type of program review board, the Software Configuration (Change) Control Sub-board (SCCSB). The SCCSB may be employed if the system under development consists of large numbers of CSCIs which might be impacted by ECPs that affect only the CSCIs (see paragraph 9.1.3.3.2).

Other decisions involve sending a request to the contractor, often using an Advanced Change Study Notice (ACSN) or contracting officer letter, to provide some additional planning/impact information to allow the program manager to decide if there is enough value in the change to warrant preparation of a formal proposal to be addressed by the CCB. To assist the program manager in making these types of decisions, the program office may use a third type of board. This board is referred to as a pre-CCB type board (in some current program offices these are called Program Event Review Boards or Business Management Boards) and is comprised of functional

representatives who can be subordinates of the CCB members. [This will depend upon the program office. Some program offices will consider the pre-CCB board as being the "worker-level" board, while others will the full CCB for this function.] These boards may also take a preliminary look at the results of the ECP/CCB review before it is presented to the CCB.

Another decision process will involve the review and evaluation of the contractor's response and compliance to action items (events) required as a result of design reviews and/or configuration audits. A Program Event Review Board may be used to determine whether the necessary actions have been satisfactorily completed. If the contractor's compliance is deemed unsatisfactory or incomplete as a result of the review, the additional work required to satisfactorily complete the action will be documented and monitored by the board until the required action has been completed. The disposition of each item considered during the board's review will be documented to clearly delineate agreements reached and/or further actions required.

Although there could be three types of boards that program managers might use to assist them in making programmatic decisions, in reality some of these functions can be accomplished by the same board. On some programs, the membership on each of these boards consists of the same individuals. [In most cases, the pre-CCB is not comprised of the same individuals as the full CCB. In fact, the members of the pre-CCB are usually those functional experts who are most knowledgeable about the proposed change. However, in most of the program offices now functioning in Aeronautical Systems Division, the CCB, SCCSB (if required), and a Program Event Review Board (different than a pre-CCB) are composed of the same members.] Each program office will have many opportunities to use these various board functions

during its system development. However, if the membership of the board is identical, a board may consider different agenda items during a single meeting that are associated with each of the three areas. [For example, the CCB may first consider ECP/CCP agenda items, then change roles to handle some PDR or CDR problems.] The following paragraphs provide OIs that relate to three boards. Some program offices may consider using all three; while others may consider preparing only one OI that incorporates the needed aspects of all three functions. The CM organization, with inputs from the program manager, will need to determine the approach based on their own specific program.

9.1.3.3.1 Configuration (Change) Control Board (CCB) OI. The CCB is the primary agency established by the program office to regulate the change management process and to be responsible for change decisions. The program office's CM organization is responsible for maintaining the CCB structure. Thus, every CM organization should provide their program office a CCB OI that describes the composition, functions, and authority of the CCB. The following paragraphs provide suggestions as to the contents of such an OI.

a. Overview - Provide a generic statement that describes the intent of the OI. In most cases, this states that the OI will define the composition, function, and authority of the <weapon system> CCB.

b. References - If there are any other program office OIs that may interrelate with this OI, list them here. Some examples would be any Change Management (Change Proposal Processing) and Specification Management OIs. Also, what regulations, standards, and pamphlets will this OI supplement?

c. Definitions - A separate section should be included that describes the different terms associated with the CCB. Some of these terms would include engineering change proposal (ECP), advanced change study notice (ACSN), contract change proposal (CCP), Requests for Deviations (RFDs), and Requests for Waivers (RFWs).

d Policy - Provide a section that describes the program office's policies for the <weapon system> CCB. Some issues that are usually discussed are:

1. Scope: That the CCB is the official [usually non-voting] board formally established by the <weapon system> program manager. It exercises complete control over all proposed changes to baselined documents that pertain to the configuration of all CI/CSCIs.

2. Authority: That the CCB membership is initially established, and will be subsequently changed, only through formal orders issued by administrative authorities at the base where the program office is located, and then only at the request of the program manager. The chairperson has the official approval or disapproval authority for the program for all changes brought before the CCB. Each member of the CCB will certify the official position of their organization by indicating either concurrence or nonconcurrence with the chairperson's decision as recorded on the CCB Directive (AFSC Form 318). Any nonconcurrence is usually required to be supported in writing.

3. Responding to Change: Discuss the program office's timing requirements for emergency, urgent, and routine changes. Within how many calendar days must these changes be brought before a board and resolved? [This type of information is also provided in a Change Management OI. However, it should be provided in this OI also so that anybody within the program office who is preparing for a CCB will be able to obtain scheduling information in the same source that provides information

concerning the function of the board. This makes it easier for program office personnel to know what is going on.]

4. Criteria for Change Approval: Discuss the program criteria for approval of change proposals. Normally, this would follow AFR 14-1, which states that changes should only be approved if they: (a) correct a deficiency; (b) make a significant effectiveness change in operational or logistics support requirements; (c) effect life-cycle cost savings or avoidance; (d) prevent slippage in approved program schedule; and (e) correct administrative errors in contractual documents.

e. Procedures - Discuss the procedures that are associated with the running of the program office's CCB. Some areas that are normally discussed are:

1. Administrative: State that the CM organization is responsible for administering the CCB. That is, they will schedule all change proposals for the board; call the CCB meetings as required; publish agendas to inform members of the date, time, and place of the meetings; provide a secretariat for the CCB; and document the results of the CCB in official minutes.

2. Agenda: Specify how many days before the meeting a formal agenda must be prepared and distributed. The agenda should list the changes to be reviewed, and the tentative order of their presentation at the CCB.

3. Meetings: Specify the normal meeting day or general plan to be followed in scheduling CCB meetings. [Usually, even if the CCB schedule is on some set time table, the CCB chairperson is given the authority to call emergency meetings for time-critical program requirements. All members, or alternates for those members, must be present at each CCB meeting, so regularly scheduled meetings will facilitate the ability to attend. If, for some reason, neither the member or alternate is available, then the

CCB can be held but officially a decision cannot be made. Given that the change proposal, or the CCB, can not be held until all members or their alternate are available, the program office may take one of two actions. The first (and probably the best approach) would be to have new orders issued identifying a member from the organization who can be present as a CCB member. The second approach (not recommended if the first approach is available), is to have the board make a decision with the member, and alternate, absent, and have one or the other sign the CCB Directive when they arrive back in the program office and are briefed about the board deliberations.]

4. CCB Presentation: Discuss the requirements for information that should be included in presenting change briefings before the CCB. [NOTE: Usually, the change request is presented by either the project manager assigned (called a change monitor) to the change, or by a CM organization assigned resource.]. The OI should include (a) standard briefing chart formats, if any, that should be used (if so, attach copies of these to the back of the OI.); and (b) what information needs to be presented to the board (a listing of the type of information should also be attached to the back of the OI).

5. Disposition of Proposed Changes: The OI should discuss the options available to the CCB chairperson for each change proposal presented. It should also require the CCB chairperson's final decision to be documented on the CCB Directive (CCBD), and each board member (or alternate) to sign the CCBD to show concurrence or nonconcurrence with the decision. The CCB chairperson is normally required to: (a) approve the proposal as written; (b) disapprove the proposal with reasons clearly stated on the CCBD; (c) approve the proposal with specific changes (it is then returned

to the contractor for corrections or revision before being finally approved and sent to the contracting officer); (d) defer action until further investigation (assigned to a specific member) is performed; and (e) defer action for changes which affect program parameters beyond the authority of the program manager (these type of changes must be referred to higher authorities for action).

6. Disposition of Decision: Define the actions to occur after the decision is made. [In most instances, the program office's CM organization is responsible for the timely forwarding of the CCBD to the contracting directorate. Discuss what should be done with the CCBD. The CCBD provides direction to the procuring contracting officer that must be passed on to the contractor.] This paragraph should also provide direction that no change shall be negotiated at a price that is in excess of the amount authorized by the CCBD. If any significant changes are made to the price or technical content of the approved proposal as a result of negotiations, an officially amended CCBD must be issued before the contractor can be authorized to proceed.

f. Membership - Discuss what offices will be required (through the issuance of official orders) to provide a member (and alternate) to the CCB. [In most cases, the program manager (or the deputy program manager or the head of projects or the engineering organization) is designated as the CCB chairperson, and each program office directorate is provided a permanent seat on the CCB. If there are any outside agencies (but not the contractor) associated with the program (e.g., supporting and using commands, safety organization, other services), they may also be provided membership on the CCB. In addition, the CM organization will normally provide a secretariat to the CCB.] Finally, discuss how changes in membership are accomplished.

g. Responsibilities - The OI should also discuss the responsibilities of each of the functionals within the program office for the conduct of the CCB. [The following paragraphs provide a suggested way to distribute the responsibilities associated with the program office's CCB.]

1. Program Manager: This person should: (a) designate the CCB chairperson, an alternate chairperson, and the CCB secretariat; and (b) provide the final resolution of, and decision on, any disputes over CCB decisions.

2. CCB Chairperson: This person should: (a) review all change proposals prior to CCB review; (b) preside over all CCB meetings; and (c) decide on the disposition of all changes presented to the CCB.

3. Directorate of Configuration Management: Some of the responsibilities of this organization are to: (a) assign a change manager/monitor for each proposed (ACSN/ECP/CCP) change; (b) provide a secretariat for the CCB; and (c) maintain the CCB file (e.g., copies of the change proposal and of the CCBD).

4. Change Manager/Monitor: This person should be responsible for: (a) providing support to the Directorate of Configuration Management on all assigned change proposals; (b) obtaining technical and/or contractual assistance, as required, to evaluate and process assigned proposals; (c) verifying availability of needed functional specialists for the change proposal prior to scheduling a CCB date; (d) conducting pre-CCB meeting (if required) and assuring comments are resolved; (e) briefing the CCB on how and why the change was generated, and all change effects; (f) providing the CCB secretariat with copies of presentation materials; (g) obtaining all "official" amendments to correct incomplete or inaccurate proposals; (h) participating in negotiations on proposals; and (i) reviewing draft minutes of CCB action.

5. CCB Secretariat: This person should: (a) receive and distribute copies of all change proposals and any amendments to the proposal; (b) distribute the CCB agenda; (c) maintain permanent record of all change proposal activity; (d) prepare CCB minutes; (e) prepare CCB Directive (CCBD); (f) forward to program manager all disputed CCBDs along with member's corresponding nonconcurrence report; and (g) provide the contracting officer the applicable CCBD for contractual incorporation.

6. Directorate of Engineering: This directorate should: (a) review the proposal for adequacy and accuracy of technical content; (b) possibly brief (if requested by change manager/monitor) and recommend a course of action to the CCB; and (c) provide the CCB secretariat with a copy of any presentation material.

7. All directorates should: (a) review and comment on change proposals when appropriate; and (b) designate a permanent, and alternate, member to the CCB.

9.1.3.3.2 Software Configuration (Change) Control Sub-board (SCCSB) OI. This type of program management board may operate as a separate entity on programs that include large number of computer software CIs, especially where the CSCI design (and changes) does not impact hardware CIs. Usually this board operates as an extension of the CCB, advising them (if the members of the SCCSB are different than those of the CCB) of the software impacts of changes affecting both hardware and software. If the program office uses the SCCSB, then the following could be included in an OI that describes the functioning of such a board.

a. Overview - Provide a statement that describes the intent of the OI. Describe the composition, function, authority, and procedures of the SCCSB and its relation to the program's CCB.

b. References - If there are any other program office OIs that may interrelate with this OI, list them here. Some examples may be either a Change Management (Change Proposal Processing) or Configuration (Change) Control Board OI. List the regulations, standards, and pamphlets that this OI supplements.

c. Definitions - A separate section may be included that describes the different terms that may be encountered by program office individuals associated with CSCIs and proposed changes to these CSCIs.

d. Policy - Provide a section that describes the program office's policies for the inclusion of a SCCSB. This policy section should state that the SCCSB will be able to act independent of the CCB only for proposed engineering changes that impact CSCIs only (and do not impact any hardware CI requirements). If hardware CIs are impacted, then define the role of the SCCSB in providing software expertise to the CCB decision-making process <and the other organizations that may be requested to provide additional inputs>. Other topics that can be discussed are:

1. Scope: Define the role and membership of the SCCSB in relation to the CCB. If it operates, then a SCCSB is usually referred to (as was the CCB) as an official, non-voting, board formally established by the <weapon system> program manager.

2. Authority: Specify the authority the Sub-board chairperson has for official approval or disapproval authority of software-only change proposals brought before the SCCSB. Specify the responsibility of each member of the SCCSB in providing their organization's official position and functional area evaluation on each change. [The members certify their official position by concurring or nonconcurring with the chairperson's decision recorded on the CCBD.]

3. Submittals: Identify the format for software-only ECPs to be submitted for action by the SCCSB. [Many program offices that use SCCSBs require that the contractor submit a shortened form of ECP as described in MIL-STD-481. The other alternative is to follow the format of MIL-STD-480, Appendix F. If this is acceptable, then a statement should be included in the OI that states this.]

4. Responding to Change: Discuss the processing (time) requirements for software change proposals that are submitted as urgent, routine (work stoppage), or routine (improvements).

e. Procedures - Discuss the procedures associated with a program's SCCSB. Areas that may be discussed in this section include:

1. Administrative: Who is responsible for administering the SCCSB? Usually, the CM organization must schedule all change actions for review; call the SCCSB meetings as required; publish agendas to inform members of the date, time, and place of the meetings; provide a secretariat for the SCCSB; and publish minutes of the meeting.

2. Agenda: Is a formal agenda required? If an agenda is required, it should list which items will be reviewed and provide a listing of the tentative order in which the proposed changes will be reviewed and discussed.

3. Meetings: When will the SCCSBs be held? Will they be a part of previously scheduled CCBs (the normal approach)? Or, will they be independently scheduled when CSCI-only proposed changes are received? As with the CCB, the SCCSB chairperson is normally authorized to convene the board at any time.

4. SCCSB Presentation: Discuss (a) what information should be provided in presenting the change to the board; and (b) if there are standard briefing chart formats that should be used (If so, attach copies of these to the back of the OI.).

5. Disposition of Proposed Changes: The OI should discuss the options available to the SCCSB chairperson for each change proposal presented to the SCCSB. The chairperson's final decision should be documented on a CCBD, which is then signed (either concurred or nonconcurred) by each board member. The chairperson can decide to: (a) approve the proposal as written; (b) disapprove the proposal with reasons clearly stated on the CCBD; (c) approve the proposal with specific changes (these are stated on the CCBD); (d) defer action until further investigation is performed; and (e) defer action for proposed changes which affect areas beyond just the software (these type of changes must be referred to the CCB for action).

6. Disposition of Decision: Discuss what happens after the decision is made. As in the CCB case, the CM organization often distributes the coordinated CCBD to all affected Government agencies. The CCBD is then used by the procuring contracting officer to issue direction to proceed, to the contractor.

f. Membership - Who should be a member of the SCCSB? Are they the same members as with the CCB?

g. Responsibilities - The OI should also discuss the responsibilities of each of the functionals within the program office for the conduct of the SCCSB. These include:

1. Program Manager: This person should designate a SCCSB chairperson and alternate, a SCCSB secretariat, and provide final resolution of problems relating to decisions made by the SCCSB.

2. SCCSB Chairperson: This person should: (a) review all CSCI change proposals prior to the SCCSB review; (b) preside over all SCCSB meetings; and (c) decide on the disposition of all changes presented to the SCCSB.

3. Directorate of Configuration Management: Some of the responsibilities of this organization are to: (a) assign a change manager/monitor for each CSCI change proposal; (b) provide a secretariat for the SCCSB; and (c) maintain the SCCSB file.

4. Change Manager/Monitor: This person should be responsible for: (a) providing support to the Directorate of Configuration Management on all assigned CSCI change proposals; (b) obtaining technical and/or contractual assistance, as required, to evaluate the change and to determine the impacts, if any, of the proposed CSCI change on the hardware CI requirements; (c) processing the CSCI proposal; (d) verifying availability of any assigned project managers or engineers for the CSCI change proposal prior to scheduling a SCCSB date; (e) briefing the SCCSB on how and why the change was generated, and all change effects; (f) providing the SCCSB secretariat with copies of all presentation materials; (g) participating in negotiations on proposals; and (h) reviewing the draft minutes of SCCSB action.

5. SCCSB Secretariat: This person should: (a) receive and distribute copies of all change proposals and any amendments to the proposal; (b) distribute the SCCSB agenda; (c) maintain permanent record of all change proposal activity; (d) prepare SCCSB minutes; (e) prepare CCBD; (f) forward to CCB chairperson all disputed CCBDs along with member's corresponding nonconcurrence report; and (g) provide the contracting officer the applicable CCBD for contractual incorporation.

6. Directorate of Engineering: This directorate should: (a) review the CSCI change proposal for adequacy, for accuracy of technical content, and for any affects

on hardware CI requirements; (b) possibly brief (if requested by change manager/ monitor) and recommend a course of action to the SCCSB; and (c) provide the SCCSB secretariat with a copy of any presentation material.

7. All directorates should: (a) review and comment on CSCI change proposals when appropriate; and (b) designate a permanent, and alternate, member to the SCCSB.

9.1.3.3.3 Pre-CCB OI. This OI is used to describe the policies and procedures that may be followed by the program office in obtaining agreement to request formal engineering, or contract/task change proposals from the contractor. As the system/CIs are under development (or in production), either a contractor or Government employee might come up with an idea for some additional work that could improve the technical capabilities of the CI and/or enhance the overall accomplishment of the program tasks. This type of idea would first be documented in an advanced change study notice (ACSN) which addresses the additional requirements or work. The pre-CCB would review the results of the program office review of the ACSN. If it decides to approve the ACSN, it will be authorizing the contractor to spend money to prepare a formal proposal (technical and pricing information) that describes in detail the requested change and its impacts.

Most programs include a process by which they also use the pre-CCB to review the results of the coordination of routine formal ECPs and CCP/TCPs to make sure they are ready for the CCB. The pre-CCB is also used by some program offices to review and evaluate the contractor's response to action items produced at the end of design reviews and/or configuration audits. The board determines whether the necessary actions have been satisfactorily completed. The following paragraphs

describe sections of an OI that can be used to discuss requirements for a pre-CCB review board.

a. Overview - State the intent of this pre-CCB process OI <to establish policies and procedures to be followed in obtaining program office approval to request technical and pricing proposals from the contractor for contract work changes>, <to be used as a "dry-run" board for routine changes being prepared for CCB discussion>, and <to review and approve/disapprove the contractor's compliance with action items assigned at either design reviews or configuration audits>.

b. References - If there are any other program office OIs that may interrelate with the OI, list them here. An example might be the CCB OI. Also, identify the regulations, standards, and pamphlets this OI is supplementing.

c. Definitions - Define any terms relating to the pre-CCB that need to be described for the program office.

d. Policy - State the purpose(s) the program manager and program office are trying to accomplish using the pre-CCB. Some areas that may be discussed are:

1. State how the pre-CCB will be used as an extension of the CCB to preview routine changes and approve their readiness to be presented to the CCB.

2. State how the board, for ACSNs, will be approving the need for obtaining the additional information; the adequacy of the description of the change required; and the schedule for submittal for the formal proposal.

3. State how the board will be reviewing the contractor's response to action items assigned at design reviews and/or configuration audits to determine whether the contractor has successfully completed the work or to define additional work required to satisfactorily complete the requested action.

e. Procedures - What procedures should be followed to accomplish a pre-CCB? Some areas that may be addressed are:

1. Administrative: Is the CM organization responsible for administering the board? Do they schedule all activities for the review?

2. Agenda: Is a formal agenda required? If an agenda is required, it should list which items will be reviewed and provide a listing of the tentative order in which the proposed changes will be reviewed and discussed.

3. Meetings: When will the pre-CCB be held? Will it be a separately scheduled meeting, or will it be a part of a previously scheduled CCB?

4. Presentation: Normally, at the pre-CCB for ACSNs, the project manager assigned to the proposed change should provide information resulting from the program office review of the ACSN and a plan for accomplishing the requested new and additional work. [NOTE: This usually requires the project manager to have held preliminary discussions with the contractor on the proposal.] For formal proposals, the project manager would present the briefing intended to be presented to the CCB. For reviewing the contractor's compliance to action items, the background, concerns, and contractor's responses are presented to the board.

5. Disposition: What happens if the ACSN is approved? Where does the ACSN go after the pre-CCB? In most cases, the board's decision, with a copy of the ACSN, is given to the procuring contracting officer who transmits a contract letter requesting the contractor to prepare a technical and cost proposal. For review/audit action items, what happens if the contractor's response to the action item is deemed unsatisfactory or incomplete?

f. Membership - Who are the members of the pre-CCB?

g. Responsibilities - The OI should also discuss the responsibilities of each of the functionals within the program office for the conduct of a pre-CCB. Some of these responsibilities should include:

1. Program Manager: This person should: (a) designate a chairperson for the pre-CCB, an alternate chairperson, and a secretariat.

2. Pre-CCB Chairperson: This person should: (a) review the ACSN, or the contractor's response to an action item, prior to the pre-CCB review; (b) preside over all pre-CCB meetings; and (c) decide on the disposition of all ACSNs, or action item responses presented to the pre-CCB for review.

3. Directorate of Configuration Management: Some of the responsibilities of this organization are: (a) assign a review coordinator for the ACSNs and/or action items to be reviewed by the pre-CCB; (b) provide a secretariat for the pre-CCB; and (c) maintain the pre-CCB file.

4. Review Coordinator: This person should: (a) act as the focal point for all matters pertaining to the ACSN or contractor's response to action item; (b) obtain technical and/or contractual assistance, as required, to evaluate ACSNs and contractor's responses; (c) conduct the meeting of the pre-CCB; (d) brief the pre-CCB on the ACSN or the contractor's response to an action item; (e) provide the secretariat with copies of presentation materials; and (f) review draft minutes of pre-CCB actions.

5. Directorate of Engineering: This directorate should: (a) review ACSN, or contractor's response to action item, for adequacy and accuracy of technical content; (b) provide review coordinator for contractor's response to design review action items; (c) possibly brief the pre-CCB and recommend a course of action; and (d) provide secretariat with a copy of all presentation material.

6. All directorates should: (a) review and comment on the ACSN or contractor's response to action item; and (b) designate a permanent, and alternate, member to the pre-CCB.

9.1.3.4 Change Management (Change Proposal Processing) OI. During the full-scale development phase, system and CI designs continue to evolve, and their baselines are established at the appropriate points. The program office must monitor and control an increasing amount of technical documentation. The CM organization can help the program office maintain control over these established baselines and evolving system/CI designs by developing and maintaining a complete change management (change proposal processing) system. The CM organization can define its change management system, describe what it will do for the program office, and delineate the change control responsibilities for all participants through the development of a Change Management OI. The following paragraphs provide suggested inputs that a CM organization may want to include in it Change Management (Change Proposal Processing) OI.

a. Overview - Provide an overview that describes the intent of the OI. Normally, this type of OI has been used to establish the policies, responsibilities, and procedures to control change proposals. This includes providing information on the initiation and processing of ACSNs, ECPs, contract change proposals (CCPs), and (if required) deviations and waivers.

b. References - If there are any other program office OIs that may interrelate with this OI, list them here. One OI that must be referenced is the CCB OI. [NOTE: Remember the CCB is an integral part of the program office's change management structure.] Identify any military standards which this OI is supplementing.

c. Definitions - Include a section that provides working definitions for program office personnel on the parts of the change management system. This section should include definitions of each of the different types of change proposals, including the difference in the types of ECPs.

d. Policy - Describe all the activities being pursued by the program office to maintain control of the established baselines. Some suggested items, which may also have been discussed in other OIs, are:

1. A general initial statement that declares that all change proposals brought to the program office will be assessed against the current program needs/deficiencies.

2. While most change proposals submitted to the program office will eventually be boarded at a CCB or pre-CCB, identify any circumstances established to allow for a proposal (e.g., a proposal that is administrative in nature) to be "handcarried" through the system.

3. Explain special applications and uses of the different types of proposals for this program, and the kind of information they should contain.

4. Define the established approval criteria from AFR 14-1 that need followed by the program. Include a change proposal content checklist that should be used to help evaluate each change. If the program office decides to use a standard evaluation checklist, this checklist should be attached to the OI.

5. Finally, the OI should identify the procedures for tracking and maintaining the status of all proposed changes received by the program office. Most program offices have developed internal data bases that are responsible for tracking all change proposals. Many have both a handwritten log and a computerized version. This section of the policy guidelines should address the information that will be maintained.

[A suggested list (not all inclusive, as each program may want to add or subtract from this list to develop its own list of "milestones") of information to include in this system is: (a) proposal number and title; (b) the idea's originator (contractor or Government); (c) date of submittal; (d) the change monitor (usually a program office functional expert or the configuration manager assigned responsibility to oversee the change proposal); (e) all reviews and technical evaluation milestones; (f) when the CCB (if required) is scheduled; (g) the result of the CCB; (h) completion date of the CCBD; and (i) date the CCBD was sent to the procuring contracting officer; and (j) date the official authorization was communicated to the contractor.]

e. Procedures - Provide information as to what steps should be performed for change proposals to be submitted through the program office. Usually this should include information about the review of routine, urgent, and emergency change proposals.

1. Somewhere early in the identification of a proposed need or deficiency, a change monitor should be assigned. The change monitor will become the focal point for processing the change within the program office.

2. This OI should discuss whether ACSNs are required and what should be included in the ACSN process. [In most instances, proposed changes will be routine in nature. To help eliminate the submittal of needless changes, most program offices have required the use of ACSNs before routine change proposals are submitted. These ACSNs may be prepared by either program office or contractor personnel. This OI should discuss what should be included in the ACSN process. In some cases, technical interchange meetings between the program office and the contractor may be

used to allow the change monitor to gather fundamental information about the change idea.]

3. List the steps that must be accomplished, either after receipt of a contractor ACSN at the program office or to complete processing of a program office ACSN (idea), before the ACSN can be approved/disapproved.

4. Similarly, when an ECP, CCP, deviation, or waiver is received by the program office, list the activities that the change monitor accomplish before, and after, the proposal is reviewed by the CCB.

f. Responsibilities - Discuss the responsibilities of the program participants who must contribute to the change management process. Some suggestions to consider are:

1. Government Plant Representative: This agency should (a) have the responsibility for concurring or nonconcurring with all engineering changes classified as Class II; and (b) review all ACSNs, and resulting change proposals, and forward comments to the program office in time for the CCB.

2. Program Manager: Designate appropriate program participants to be members of the CCB and provide decisions on proposed changes, affecting program work tasks and baselines.

3. Directorate of Configuration Management: This directorate is the office of primary responsibility and aids the change monitor in maintaining control of the change proposal's flow through the program office. Some examples of its responsibilities are: (a) to develop a change tracking system to provide the program office the traceability of all change proposals; (b) to process and submit to the contractor all program office requests (ACSNs) for change proposals; (c) assign a change monitor to all proposed

changes; (d) to assist the change monitor in establishing a schedule of events for the timely review of change proposals; (e) to receive change proposals, establish and maintain change tracking information, and coordinate the proposal for the change monitor; (f) to monitor and maintain control of the internal review processes to ensure efficient change proposal processing and coordination; (g) to ensure that all information required to evaluate the change proposals is available and to assist in acquiring any missing information; (h) to schedule and convene a CCB for each change proposal; and (i) to record CCB results on the CCBD and provide the contracting organization with the instructions that must be passed on to the contractor.

4. Change Monitor: This person should be the program office's focal point for the evaluation of the proposed change. Normally, the change monitor is assigned from either the projects or the engineering directorate, but the configuration manager may also act as the change monitor. This individual must work closely with the CM organization to make sure that adequate information has been provided and that a complete review has been accomplished. Some tasks that need accomplished include: (a) review change documentation for content and format; (b) evaluate the problem and proposed solutions and the coordination inputs from participating functional activities to determine impacts on system performance, program cost and schedule, technical data, established baselines, and testing procedures; (c) ensure that necessary funding is available; (d) participate at all meetings concerning the proposed change; (e) present the change proposal briefing at the CCB; and (f) participate in fact-finding and negotiations leading to contractual authorization.

5. Directorate of Engineering: This group should (a) evaluate the change proposal and provide technical recommendations to the change monitor and CM

organization; (b) participate in technical information meetings and CCBs that discuss the proposal; and (c) recommend either approval or disapproval of the change.

6. Directorate of Contracting: This group should (a) review the change proposal for contractual impacts and any in-scope/out-of-scope determinations; (b) determine appropriate contract instrument (usually a supplemental agreement or change order) to implement the approved change in time to meet the established need date; (c) ensure negotiators are aware of comments on CCBD; and (d) contractually implement all CCB decisions.

7. Directorate of Program Control: This group should provide the change monitor with information on the cost breakdown associated with the proposed change. They should also review the change for impact on contract cost and schedule and verify funds availability.

8. All program participants should review the proposals for impacts in their areas of expertise and recommend whether the proposal should be approved or disapproved.

9.2 CM Requirements for the Contractor.

In addition to the activities being pursued by the CM organization within the program office, another aspect of a successful CM program is contractually requiring the contractor to perform needed CM activities. To ensure that the contractor is aware of those actions being required, the CM personnel incorporate tasking paragraphs into the Statement of Work (SOW) addressing those actions it is requiring. The following paragraphs address those areas of CM, for a program entering full-scale development, where the contractor is normally required to perform some type of task. As with OIs, not all programs will find it necessary to use all of these suggested tasking paragraphs.

However, because of a particular program's circumstances, it may be required to provide tasking paragraphs not addressed below. However, the following should be reviewed as a starting point from which the configuration managers could make decisions about the tasks to include in the SOW. [NOTE: AFSCP 800-6 requires the use of an interactive SOW generator, such as the Computer Generated Acquisition Document System (CGADS), in developing the SOW. The CM organization should access the CM portion of CGADS in addition to using this handbook.]

This SOW is used to define the work efforts required from the contractor to support the program during the full-scale development portion of the system acquisition life cycle. A clear statement of contract work tasks is a prerequisite for defining and achieving program goals and objectives. The SOW provides the basic framework for this effort. It must be prepared to specify basic management responsibilities and minimum program requirements. For full-scale development, the CM portion of the SOW is used to define specific technical management activities required of the contractor. In particular, the SOW paragraphs used should provide for the implementation of a CM program by the contractor (tailored to the scope of the specific program) that finalizes detailed specifications and exercises configuration control over the established functional baseline and provides for configuration control over each allocated baseline as it is established.

The SOW requires tasks to be performed. Sometimes, as a result of the work task, information/data/documents may be generated. In order to contractually require delivery of plans, reports, or other information, the SOW paragraph must be complemented by a Contract Data Requirements List (CDRL) item. The CDRL identifies the data items that a contractor must deliver and references the Data Item

Descriptions (DIDs) that define the data item contents and formats. The CDRL will also establish submission dates, numbers of copies, distribution, inspection and acceptance criteria, and other related information. <NOTE: Given today's requirements to minimize data costs, the person requesting the data should review the applicable DID content to specifically tailor out data not required for this program.>

The following paragraphs describe and outline typical SOW paragraphs (with the appropriate DID number provided) that should be considered by configuration managers for inclusion in the full-scale development SOW. These paragraphs are only suggestions. Each program should review and tailor these for their own application.

9.2.1 Configuration Management Plan (CMP).

The CMP should be required as a contractually deliverable data item and as a general guideline for the contractor's CM practices, but it usually should not be incorporated into the contract as a contractually binding document. The difference between these two approaches is that, as a contractual document, anytime the specific information/practices called for by the CMP need revision, a formal (sometimes costly) contract change will be required. As a guidance document, the practices cited in the CMP may be revised through a resubmission to the program office without requiring a formal contract change; while not specifically contractual in nature, these practices still provide firm guidelines for the contractor's CM effort.

The CMP should describe the contractor's organizational responsibilities and procedures to be used for implementing the CM requirements stated in the contract. It may be used by the program office during source selection to evaluate the contractor's approach and methods, and then, after contract award, it can be used to monitor and evaluate the contractor's CM performance. To facilitate the development of a CMP,

the program office will normally require that MIL-STD-483 (Appendix I) be placed on contract, tailored for their particular program. The CMP for full-scale development may be a revision to the document prepared during concept demonstration/validation, or it may be a newly prepared document.

However, prior to the contractor preparing a deliverable CMP, the program manager (based on inputs from the CM organization, as well as from other affected functionals) must decide if such a document should be required to be submitted. For many programs, this decision may be driven by the use of the CMP during the previous phase of the system acquisition life cycle. If so, then unless a major change has occurred in the program's acquisition strategy, the CMP will probably be used (not used) as it was in the earlier phase. But, if the program is directly entering full-scale development, if major changes in the acquisition strategy have been directed, or if the initial decision to require/not require a CMP was postponed, then the program office will have to decide whether or not to require the contractor to prepare a deliverable CMP. Important considerations in the decision are the contractor's previous experience with Government programs and the technical complexity of the current program. [For those contractors who have designed, developed, and managed acquisition programs in the past and who have maintained a successful CM program, it might be cost effective to require the contractor to use its own internal-developed CM program without requiring a deliverable CMP, unless the technical complexity of the current program is rated as high. On the other hand, if the contractor selected for the development program is relatively new to the Government system acquisition process, then if the technical complexity of the proposed system development is only moderate, it may be advisable to require the contractor to submit a deliverable CMP to insure that

the correct procedures will be in place to establish and maintain a configuration management program.]

If it is decided to require the contractor to prepare and submit a deliverable CMP, another decision that the program office must decide is whether or not they will want the contractor to prepare a separate Software CMP in addition to a Hardware CMP. [This course of action should only be followed in extreme cases, and only with those programs that are composed almost entirely of technically complex CSCIs and a low number of hardware CIs. Otherwise, it is more economical for the program office to request its contractors to include their CM practices for the CSCIs as a part of its overall CMP and/or to include their software CM practices in the Software Development Plan (SDP). The SDP describes the contractor's overall plan for developing software and supporting documentation, and it also provides the procedures for managing and controlling the software development process.]

The following paragraphs suggest SOW taskings to deliver a CMP as either a new submittal or a revision to a deliverable item from the previous system acquisition life cycle phases.

a. Typical SOW Paragraph (new tasking) - The contractor shall identify, within its program management structure, a configuration management organization responsible for performing the requirements of the CM processes of configuration identification, configuration audits, change management, and configuration status accounting during the full-scale development phase of system development. The contractor shall separately document the CM practices to be utilized for this program. The CMP, as approved by the acquiring agency, shall be used as a guide in determining compliance.

b. Typical SOW Paragraph (revision of CMP) - The contractor shall continue the CM program established during concept demonstration/validation. The contractor shall maintain and update its CMP as required to include any new procedures/requirements that arise during full-scale development.

c. DIDs - The current DID for a deliverable CMP is DI-CMAN-80405 (replacing DI-E-3108).

9.2.2 Configuration Identification Requirements.

In addition to being tasked to establish the overall CM policies, plans, and procedures within its organization to meet its CM requirements, the contractor is also normally tasked to meet specific requirements for each of the CM sub-processes. With regards to configuration identification, the contractor is to assist the program office in the establishment of baselines, the preparation and submittal of technical documentation (e.g., specifications and drawings), and the naming and identification numbering of the system, segments, and CI/CSCIs.

(For most major programs entering full-scale development, the contractor should already have prepared and submitted (and the program office authenticated) the System Specification as a part of the successful completion of the System Requirements Review(s), establishing the functional baseline. In this case, the SOW will contain tasking paragraphs requiring the contractor to maintain and comply with this previously established baseline. However, in some instances (not the normal case for most major programs), the program office may not have established a functional baseline prior to entering full-scale development. If this occurs, then the CM organization must include a tasking paragraph for the contractor that requires the submittal of a final draft System Specification for authentication and establishment as

the functional baseline. Both of these cases are provided for in the paragraphs that follow. If the functional baseline (and the System Specification) has already been established, then the program office may want to include paragraph 9.2.2.1. Or, if the functional baseline has not already been established, then the program office may also want to include paragraph 9.2.2.2.]

The following paragraphs describe various SOW taskings that may be required of the contractor to accomplish the configuration identification function. [With the exception of the alternatives for the functional baseline just discussed, these paragraphs have been used in some form or another by most programs entering full-scale development.]

9.2.2.1 Previously Established Functional Baseline. If the program office has already established the functional baseline (and authenticated the System Specification) prior to the program entering full-scale development, then the following paragraph may be used in the SOW.

a. Typical SOW Paragraph - The <weapon system> functional baseline is as documented in the <weapon system> Specification <provide the specification number and authentication date>. The <weapon system> Specification shall be updated and shall incorporate a specification tree identifying all hardware and computer software configuration items. The updated <weapon system> Specification shall be reviewed and approved by the procuring activity. After approval of the procuring activity is granted, the new specification shall supersede the <weapon system> Specification and will be noted in <the appropriate section of the contract>.

b. DID and Rationale for Use - The current DID for the deliverable System/System Segment Specification (SSS) is DI-CMAN-80008A (which has replaced DI-E-3101 used

on older contracts). The SSS specifies the functional, performance, and interface requirements for a system or for a segment of the system. As the functional baseline, the SSS provides the Government with a vehicle for conveying system requirements to the contractor, and it provides an overview of the system to trainers, maintainers, users, and supporters of the system.

9.2.2.2 <u>Establishing the Functional Baseline.</u> If the program has not established a functional baseline prior to starting full-scale development, the configuration manager may want to use a SOW tasking similar to the following. [The SOW should probably also include the previous SOW tasking paragraph to allow updates to the specification once it is authenticated.]

a. Typical SOW Paragraph - The contractor shall prepare the <weapon system> Specification in accordance with MIL-STD-490, Appendix I. The <weapon system> Specification shall be updated and shall incorporate a specification tree identifying all hardware and computer software configuration items. The functional baseline shall be established after authentication of the <weapon system> Specification by the procuring activity.

b. DID and Rationale for Use - The current DID for the deliverable System/System Segment Specification (SSS) is DI-CMAN-80008A (which has replaced DI-E-3101 used on older contracts). The SSS specifies the functional, performance, and interface requirements for a system or for a segment of the system. As the functional baseline, the SSS provides the Government with a vehicle for conveying system requirements to the contractor, and it provides an overview of the system to trainers, maintainers, users, and supporters of the system.

9.2.2.3 Allocated Baseline(s). As the program progresses into full-scale development, the contractor will be allocating the system requirements to the system's CI/CSCIs. The SOW should contain a tasking paragraph that requires the contractor to establish the allocated baseline for each CI/CSCI at the appropriate times. This requires the contractor to submit the Type B Specifications to the program office for authentication. The following SOW paragraph may be used to request these specifications, and, through them allow establishment of the allocated baseline(s) for the CI/CSCIs of the system. A paragraph of this type should be included in all full-scale development SOWs. [The CM organization must make sure it understands the acquisition strategy of the program. If the program is using a two-part specification, then the suggested changes discussed in the paragraph may have to be used.]

a. Typical SOW Paragraph - For each hardware configuration item (HWCI) and computer software configuration item (CSCI) to be developed, the contractor shall document the requirements in accordance with MIL-STD-490, Appendices II, III, IV, V, and VI <the use of the Appendices is tailored based upon the type of program being developed>. Upon satisfactory completion of a joint Government/contractor specification review for each draft specification, which will occur as soon as possible after Government comments are forwarded to the contractor, the Government will authenticate the specification and establish the allocated baseline for that CI.

[Alternative 1: If the program office has decided that phased baselining will be used, then the SOW and CDRL should read such that the higher-level CIs (a listing of which has been agreed upon by both the contractor and the Government) shall be authenticated prior to PDR. The lower-level CI specifications shall not be authenticated or baselined until the CI/CSCI's Functional Configuration Audit.]

[Alternative 2: The program office may also want to use a two-part specification approach. If this is the planned approach, then the wording of this paragraph may be altered such that after discussing the Appendices of MIL-STD-490, the SOW paragraph should continue as: The development/requirements specifications shall be prepared as Part I of a two-part specification as discussed in MIL-STD-490 (paragraph 3.1.4).]

b. DIDs and Rationale for Their Use - The following paragraphs provide the DID numbers and rationale for each type of specification that can be used for establishing the allocated baseline for a CI or CSCI. The use of any, or all, of these specifications is dependent upon the individual program.

(1) Configuration Item Development Specification: The current DID for this deliverable specification is DI-E-3102A. This DID provides for the delivery of Prime Item, Critical Item, and Non-Complex Item Development Specifications that allow for the establishment of all performance, design, development, and test requirements for all CIs developed under the contract. The Government and contractor will use these specifications as the contractual source of the requirements which the CIs must meet. Without these specification, there might not be any contractual basis for the development of the CIs.

(2) Software Requirements Specification: The current DID for this deliverable specification is DI-MCCR-80025A (which replaced DI-E-30113). This DID should be required for all procurement programs with any CSCIs identified. The DID provides for the delivery of a specification that establishes the functional, performance, and quality assurance requirements for each CSCI. The contractor will use this specification as the basis for the development and formal testing of the CSCI.

(3) Interface Requirements Specification: The current DID for this deliverable specification is DI-MCCR-80026A. [This DID should not be used on every program. When the system under design contains only a few interfaces between HWCIs and CSCIs, it is more beneficial (cost and programmatically) to include the interface requirements as a part of the Software Requirements Specification.] The DID allows for the delivery of a specification that provides the description of the specific requirements for the interfaces between the HWCIs and CSCIs.

9.2.2.4 Product Baseline(s). Some programs may be able to establish the product baseline(s) of the software CIs and of some of the lower-level CIs during full-scale development. Or, depending upon the technical simplicity of the proposed design, or if the production contract is to be competed, the program office may want to also have some of its higher-level CIs product baselines established during full-scale development. If this is the strategy being pursued by the program office, then the CM organization should make sure that an appropriate SOW tasking paragraph is included to allow for this contingency. The following suggested paragraphs would provide for the establishment of some, or all, of the product baselines.

a. Typical SOW Paragraph - For each hardware configuration item (HWCI) and computer software configuration item (CSCI) that is to be developed and produced as a part of <weapon system>, the contractor shall document the detail design requirements in accordance with MIL-STD-490, Appendices VII, VIII, IX, X, XI, XII, and XIII <the use of the Appendices is tailored based upon the type of program being developed>. Upon satisfactory completion of a physical configuration audit for each draft specification, the Government will authenticate the specification. Process and

Materials Specifications shall be prepared in accordance with MIL-STD-490, Appendices XIV and XV.

[Alternative: If the program office is pursuing the two-part specification, then the above SOW tasking paragraph should be altered such that after discussing the MIL-STD-490 Appendices with the Type C Specifications, the paragraph should continue as: The product specification shall be prepared as Part II of a two-part specification in accordance with MIL-STD-490, paragraph 3.1.4.]

b. DIDs and Rationale for Their Use - The following paragraphs provide the DID numbers and rationale for each type of specification that can be used for establishing the product baseline for a CI/CSCI. The use of any, or all, of these specifications is dependent upon the individual program.

(1) Configuration Item Product Fabrication Specification: The current DID for this deliverable specification is DI-E-3103A. This DID orders a specification that provides the detailed design, test, manufacturing, and acceptance requirements for all CIs developed under the contract.

(2) Configuration Item Product Function Specification: The current DID for this deliverable specification is DI-E-3132. It orders a specification that establishes the requirements for major items of commercial equipment acquired under the contract.

(3) Inventory Item Specification: The current DID for this deliverable specification is DI-E-3105. It is used to order a specification that covers the requirements for inventory items available to the Government. The specification is required to identify Government furnished equipment, and modifications to this equipment.

(4) Software Product Specification: The current DID for this deliverable specification is DI-MCCR-80029A. This DID orders a specification that provides the detailed technical description of the CSCI, and includes the Software Design Document (called for by DI-MCCR-80012) and the CSCI's source code listings.

(5) Process Specification: The current DID for this deliverable specification is DI-E-3130. This DID orders a specification that will define the details of a peculiar treatment or process that is critical to the manufacture of the system under development.

(6) Material Specification: The current DID for this deliverable specification is DI-E-3131. This DID orders a specification that will define the details of the raw materials that are critical to the fabrication of the system.

9.2.2.5 Specification Maintenance. Once a baseline is established, the program office must ensure that the contractor is properly controlling and maintaining those specifications for which it is responsible. This requirement is levied upon the contractor regardless of which baseline(s) has been established. The following paragraph is suggested to task the contractor to perform this activity.

a. Typical SOW Paragraph - Once baselined, all specifications shall be maintained under the contractor's configuration management procedures. The contractor shall maintain all specifications for the system, and its configuration items, that are identified on the specification tree in the <weapon system> System Specification. Maintenance of specifications pertains to specification change notices, specification change pages, and specification revisions. All specifications which identify and document the functional and allocated configuration identifications shall be maintained by the

contractor in accordance with MIL-STD-480, MIL-STD-490, and <if applicable> DOD-STD-2167 as guidelines.

b. DID and Rationale for Use - These is no DID for this tasking since there is no requested deliverable specifically asked for from the contractor concerning this task. The DID that covers the delivery of specification change notices and engineering change proposals is included in the Change Management section of the SOW. [NOTE: There was a DID (DI-E-3106) associated with this task, but it has been canceled.] The basic DID for each major type of specification would be used to obtain revisions.

9.2.2.6 Engineering Drawings and Associated Lists. In addition to providing specifications that describe the configuration identification requirements of each CI/CSCI, the contractor must be required to provide a technical data package that includes engineering drawings and associated lists. The individual in the program office who is normally responsible for managing the acquisition and documentation of the engineering drawings and associated lists is the engineering data management officer (EDMO). [Many times the EDMO responsibilities are fulfilled by a configuration manager/specialist. For this reason, the configuration manager should be aware of the requirement to task the contractor in this area.] The EDMO must review MIL-T-31000 and DOD-STD-100 to determine what kind of technical data package, and which type of drawings and associated lists, should be included in the tasking. This individual must also insure that the engineering data requirements are specified in the SOW, CDRL, and, if required, in any special provisions of the contract. The following SOW paragraph is suggested to start this process, but the actual wording of each program's

SOW paragraph (and DID requirements) should be written by (or at the very least, coordinated by) the program office's EDMO.

a. Typical SOW Paragraph - The contractor, and all appropriate subcontractors and vendors, shall document the engineering design for the <weapon system> and all its support equipment. The contractor shall assemble the engineering data into <NOTE: Here is where the program office must decide which level of technical data package, from MIL-T-31000, should be required.> a complete technical data package and maintain that package to the current configuration being tested. Engineering drawings required shall be as specified in DOD-STD-100. The contractor shall maintain a master set. The master set is the property of the Government and shall be released to the Government upon the contractor receiving direction from the procuring contracting officer (PCO).

b. DIDs and Rationale for Their Use - The current DIDs for these deliverable items are DI-DRPR-81001 (Conceptual Design Drawings and Associated Lists), DI-DRPR-81002 (Developmental Design Drawings and Associated Lists), DI-DRPR-81000 (Product Drawings and Associated Lists), and others (that discuss the remaining types of technical data packages); all are listed in Section 6.3 of MIL-T-31000. They allow for the Government to obtain the engineering data required to perform management, engineering, test, maintenance, modification, competitive reprocurement of spares, and other logistics functions. The level of drawing available is also partly dependent on the phase of the system acquisition life cycle the program is entering. Although the complete package is not normally delivered during full-scale development, the program office will normally take delivery of samples for use in "In-Process Reviews" of the drawings.

9.2.2.7 Request for Nomenclature. The contractor must be required to request nomenclature assignment for each new developed system, CI, and CSCI under the contract. MIL-STD-490 requires that nomenclature be used on all specification covers; on all drawings; and on all maintenance, user, diagnostic, and operator's manuals. The DD Form 61 is used to request an assignment, cancellation, revision, or reinstatement of nomenclature. The following paragraph provides a suggested SOW tasking paragraph.

a. Typical SOW Paragraph - The contractor shall request the assignment of nomenclature for newly designed configuration items in accordance with MIL-N-7513 and MIL-STD-1661. The assigned nomenclature shall be applied to the configuration item nameplate, specifications, drawings, and other applicable data pertaining to the nomenclatured item. The contractor shall not privately assign nomenclature descriptions; only official nomenclatures shall be used. At a minimum, those items listed on the specification tree shall be provided with nomenclature assignments.

b. DID and Rationale for Use - The current DID for this deliverable item is DI-E-7194. As stated above, nomenclature is required on all documentation that identifies a configuration item.

9.2.2.8 Computer Program Identification Number (CPIN). This is required to identify the CSCI before and after the system becomes operational. The contractor requests and obtains the CPIN from the procuring activity, who in turn has received the CPIN from the Oklahoma City Air Logistics Center. The CPIN tasking must be required of any contractor that is developing, and delivering, a CSCI with the system.

a. Typical SOW Paragraph - For each CSCI, or version of a CSCI that is being developed under this contract, the contractor shall obtain from the procuring activity a

Computer Program Identification Number (CPIN). All CSCIs and the associated documentation will be numbered using the CPIN system.

b. DID and Rationale for Use - The current DID for this contract deliverable item is DI-E-3162B. This DID allows for an Air Force number to be used on all CSCI media and on the title pages of related documentation and technical manuals.

9.2.2.9 <u>Identification and Numbering of Hardware and Documentation.</u> The contractor must also be required to identify, number, and mark all hardware CIs, and their components and associated documentation which will require configuration control or will require a follow-on spares procurement effort. This identification is used to provide the program office visibility of the affected hardware CIs (and their components) developed by the contractor, subcontractors, vendors, and/or suppliers. The following paragraph is suggested to task the contractor to perform this identification requirement.

a. Typical SOW Paragraph - The contractor shall identify documentation (e.g., specifications and drawings), configuration items, and their component parts to be delivered under this contract in accordance with the requirements of MIL-STD-483, Appendix IX.

b. DID and Rationale for Use - Unlike the CPIN requirement, there is no DID for this tasking since the deliverable items are requested in the respective specification and engineering drawing paragraphs.

9.2.3 <u>Configuration Audits Requirements.</u>

Similar to requiring configuration identification taskings of the contractor, the program office must also require taskings for contractor support of the configuration audit sub-process. In this area, for full-scale development, the contractor is normally

requested to assist the program office with the functional configuration audit(s) for each CI/CSCI under development. Another audit that the program office may need to employ is the functional system audit. This would be used to ensure that the system requirements, not verified as each CI/CSCI FCA was accomplished, have been met.

In addition, depending upon the acquisition strategy being pursued by the program office, they may also want to begin (or in some instances, complete) physical configuration audits (PCAs) on some (or all) of the system's CI/CSCIs and their components. [Most programs use the same contractor for the full-scale development and production efforts. In this instance, the PCAs are conducted on an operational unit during the production phase of the system acquisition life cycle. However, in some cases, the program office may have decided that the production contract will be competed. If this occurs, then during full-scale development, a PCA will need to be performed on prototype units for each CI/CSCI, and a PCA will also be performed later during the production phase on an operational unit produced by the appropriate contractor. Due to their development cycle, CSCIs may have a PCA performed during full-scale development regardless of the acquisition strategy.] Whatever audits are required as a part of the full-scale development effort, they must be reflected in the SOW tasking paragraph.

9.2.3.1 <u>Functional Configuration Audit.</u> The following SOW paragraph is suggested for the program that will perform a FCA on one (or, in some instances, more than one) configuration item as it is ready to have its development process validated.

a. Typical SOW Paragraph - A FCA shall be accomplished to validate the development of each CI/CSCI of the <weapon> system. The contractor shall schedule and support the Government in conducting a FCA on each new or modified

configuration item in accordance with MIL-STD-1521, Appendix G. The FCA shall be conducted jointly by both the procuring agency and the contractor. Planning for each audit, and the activities during each audit, shall be documented by the contractor.

b. DIDs and Rationale for Their Use - The current DIDs that are applicable for this tasking are DI-A-7088 and DI-A-7089. These are the DIDs for conference agenda and conference minutes, respectively. These will allow for planning prior to the audit and for the documentation of discussions which took place, and decisions which were reached, as a result of the audit. [NOTE: If other audits, or design reviews, are accomplished during this phase, these data items need only be ordered once in the CDRL.]

9.2.3.2 <u>Functional System Audit.</u> The functional system audit (FSA), if required, is normally a full-scale development task. If the program office wants to include a FSA at the conclusion of the FCAs to ensure that the system, when all its parts are combined, meets all of its specified requirements, this is usually included as a part of the full-scale development SOW, even though the FSA may be conducted after the program has entered the production/deployment and operational support phase of the system acquisition life cycle. In order for the program office to include such a task in its full-scale development SOW, the following paragraph is suggested.

a. Typical SOW Paragraph - After completion of the FCAs, the contractor shall also schedule and support the Government in conducting a functional system audit (FSA) in accordance with MIL-STD-1521, Appendix I. The FSA will be conducted to validate those system requirements that could not be validated through the accomplishment of the CI/CSCI FCAs. Planning for the audit, and activities during the audit, shall be documented by the contractor.

b. DIDs and Rationale for Their Use - The current DIDs that are applicable for this tasking are DI-A-7088 and DI-A-7089. These are the DIDs for conference agenda and conference minutes, respectively. These will allow for planning prior to the audit and for the documentation of discussions which took place, and decisions which were reached, as a result of the audit.

9.2.3.3 Physical Configuration Audit. Normally, the PCA(s) is conducted during the production phase of the system acquisition life cycle, not during full-scale development. However, if the program is pursuing an acquisition strategy that will require a PCA to be performed during full-scale development (either for its CSCIs or on prototype units when the production contractor has not yet been selected), then the following paragraph is provided as a suggested approach to task the contractor.

a. Typical SOW Paragraph - The contractor shall schedule and assist the procuring agency conduct a physical configuration audit (PCA) <on a prototype unit for each new or modified hardware configuration item> <on the first deliverable program of the CSCI> in accordance with MIL-STD-1521, Appendix H. The PCA shall be conducted jointly by both the procuring agency and the contractor to verify that the CI's <CSCI's> documentation actually represents the "as-built" <"as-coded"> configuration. After successful completion of this audit, the product baseline shall be established. Planning for each audit, and the activities during each audit, shall be documented by the contractor.

b. DIDs and Rationale for Their Use - The current DIDs that are applicable for this tasking are DI-A-7088 and DI-A-7089. These are the DIDs for conference agenda and conference minutes, respectively. These will allow for the documentation of

discussions which took place, and decisions which were reached, as a result of the audit.

9.2.4 Change Management.

Once a baseline has been established the program office must ensure that the contractor is indeed controlling these baselines. In addition, after the contract has been approved, and signed, by the contractor and the program office, this "baselined" contractual document must also be controlled. To ensure that no changes are made to either the technical or contractual documentation associated with the program without the approval of the program office, the following tasking paragraphs should be included in the full-scale development SOW.

9.2.4.1 Advanced Change Study Notices. During the development phase, if the program office wants to use the generation of ACSNs prior to the submittal of any routine change [and all programs should consider this approach as almost mandatory], the contractor must be tasked to perform/use this management approach. The following paragraph suggests a way to implement this requirement in the SOW.

a. Typical SOW Paragraph - Prior to starting any work or effort on a proposed routine engineering or contract change proposal, the contractor shall first initiate an Advanced Change Study Notice (ACSN) to obtain the Government's agreement that the proposal should be processed. The ACSN must be approved to initiate the detailed change preparation. An ACSN is not required for proposals that are considered urgent or emergency.

b. DID and Rationale for Use - The current DID for this requirement is DI-E-3127A. It is used to assist the program manager maintaining cost control of contractor efforts

relative to the generation of engineering and contract change proposals. It is a way to avoid excessive preparation costs for undesirable changes.

9.2.4.2 Engineering Change Proposals, Notices of Revision, and Specification Change Notices. Once the functional baseline has been established, and as each CI/CSCI's allocated and product baselines are established, the contractor must be responsible to maintaining control of these baselines and the technical documentation that comprise the corresponding configuration identification. The following paragraph suggests a way to task the contractor to maintain this required control.

a. Typical SOW Paragraph - All baselines that have been established during earlier contracts, or that will be established during the existence of this contract, shall be controlled in accordance with MIL-STD-480. Any changes requested to be made to these baselines should be documented, and controlled, using a Class I Engineering Change Proposal (ECP). For routine Class I ECPs, the contractor shall first obtain the procuring agency's authorization to prepare the proposal using an ACSN describing the proposed change. The contractor shall submit cost (consistent with contract type) and other affected program information, as a part of the ECP. All Class I ECPs shall be submitted to the procuring agency for review and approval. The contractor shall document the impacts to the baselined specification in Specification Change Notice(s), including change pages in the same form and format of the affected document/ specification. In those cases where the ECP will affect a specification that is not maintained by the submitting contractor, then the contractor must document the impacts on a Notice of Revision.

b. DIDs and Rationale for Their Use - The current approved DIDs that affect the ECP process are DI-CMAN-80639 (ECPs), DI-CMAN-80643 (SCNs), and DI-CMAN-

80642 (NORs). The ECP DID is used by the program office to ensure that the data required to analyze performance, time, and cost benefits of proposed changes is provided with the ECP. The SCN DID is used to allow the program office to ensure that the contractor is addressing all the impacts of proposed changes to the specifications after the baselines are established. Finally, the NOR DID is used to ensure the program office that the contractor will include changes to specifications outside of its control when proposing engineering changes.

9.2.4.3 Deviations and Waivers. The program office must also provide tasking to the contractor that provides direction on the use of deviations and/or waivers, if required, during the development of the system or its configuration items. [Deviations and waivers are primarily associated with production units delivered as a part of a production contract. However, there are instances where prototypes with known deficiencies are accepted using deviations.] The following is a suggested SOW paragraph to accomplish this tasking.

a. Typical SOW Paragraph - The contractor shall initiate requests for deviations and waivers in accordance with MIL-STD-480. Minor deviations and waivers shall be submitted to the Government plant representative for concurrence or nonconcurrence with the assigned classification category. Major deviations and waivers shall be submitted to the procuring agency for review and approval.

b. DID and Rationale for Use - The current approved DID for this SOW tasking is DI-CMAN-80640. This DID will allow the contractor the option to obtain authorization to depart from a particular performance or design requirement of a specification.

9.2.4.4 Contract/Task Change Proposal. In addition to allowing for the tasking of the contractor to control the technical baselined documents, the program office must also be prepared to task the contractor to control other non-technical contractual documents. The following is a suggested SOW paragraph to meet this requirement.

a. Typical SOW Paragraph - The contractor shall process (after receiving approval of an ACSN) a Contract/Task Change Proposal (CCP/TCP) in accordance with MIL-STD-483, paragraph 3.14, whenever a non-engineering contract change is being proposed. Each CCP/TCP shall include a firm price proposal impact to the contract price.

b. DID and Rationale for Use - The current DID that provides for this tasking is DI-A-3020B. This DID allows the contractor to submit proposed changes to the contract in areas other than those impacting the technical baselines.

9.2.5 Configuration Status Accounting.

Finally, the contractor must be required to develop and maintain an information data base that will provide the needed information about all CI/CSCIs designed, developed, and delivered to the procuring agency. This data base should provide, at a minimum, the configuration documentation and identification numbering information and information about the implementation of approved changes to the technical documents. The following is a suggested SOW paragraph that can be used to task the contractor.

a. Typical SOW Paragraph - The contractor shall develop and maintain an information data base that is capable of: (1) describing the technical documentation that comprises the appropriate configuration identifications; (2) providing all essential CI and CSCI identification numbering data; (3) describing all contractual (non-technical)

documentation; (4) providing a listing of all proposed changes, and the status of approved changes not yet completely implemented, to all program documentation; and (5) documenting the specific configuration of all items that are used during testing activities.

b. DID and Rationale for Use - The current approved DID for this tasking paragraph is DI-E-3133. This DID provides the means by which the Government can obtain configuration status accounting records and information from the contractor on various program management concerns.

Bibliography

1. Collins, Larry R., Software Configuration Manager. Personal Correspondence. DCS, Integrated Engineering and Technical Management, Wright-Patterson AFB OH, 7 August 1990.

2. Dean, William A. Class handout distributed in SYS 028, Introduction to Configuration Management. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, October 1987.

3. -----. Class handout distributed in SYS 228, Applied Configuration Management. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 1 October 1987.

4. -----. "We Still Worry About Configuration Management," AFIT/LSY WPAFB OH (unpublished).

5. Department of the Air Force. Acquisition Management: Configuration Management. AFSC Pamphlet 800-7. Washington: HQ USAF, 1 December 1977.

6. -----. Configuration Management. AFR 14-1. Washington: HQ USAF, 1 December 1988.

7. Department of Defense. Configuration Control - Engineering Changes, Deviations and Waivers. Military Standard 480B. Washington: Government Printing Office, 15 July 1988.

8. -----. Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs. Military Standard 483A. Washington: Government Printing Office, 4 June 1985.

9. -----. Configuration Status Accounting Data Elements and Related Features. Military Standard 482A. Washington: Government Printing Office, 1 April 1974.

10. -----. Technical Reviews and Audits for Systems, Equipments, and Computer Software. Military Standard 1521B. Washington: Government Printing Office, 4 June 1985.

11. Haynes, James, Director of Configuration and Data Management. Personal Correspondence. DCS, Integrated Engineering and Technical Management, Wright-Patterson AFB OH, 16 August 1990.

12. Shaw, Thomas E. "Back to Basics," (Keynote Address) Report of the 21st Annual Workshop: Computer Resources and Data and Configuration Management Workshop. Savannah GA, November 2-6, 1987.

13. "Transition From Current to Automated Environment," (Sub Panel B-3) Report of the 22nd Annual Workshop: Computer Resources and Data and Configuration Management Workshop. Minneapolis MN, September 26-30, 1988.

Vita

Captain James E. Corbin was born on 24 July 1959 in Pittsburgh, Pennsylvania. He graduated from South Hills High School in Pittsburgh in 1977 and attended the United States Air Force Academy, graduating with a Bachelor of Science in Chemistry in May 1981. Upon graduation, he received a regular commission in the USAF and served his first tour of duty at Langley AFB, Virginia. While there, he was a Programs and Plans manager for the Tactical Air Command's aircraft training ranges. He was then selected for the Air Force's Engineer Conversion Program and attended the University of Maryland from August 1983 to May 1985 when he graduated with a Bachelor of Science in Aeronautical Engineering. He was then assigned to Aeronautical Systems Division where he began as an aircraft performance engineer on such programs as the F-16, A-7 Plus, LANTIRN, and F-15. He went on to become a Chief Flight Systems Engineer on the Combat Talon II aircraft. Finally, he was the Chief Flight Systems Engineer on the joint service (with the US Navy) Tacit Rainbow missile program. He entered the School of Systems and Logistics, Air Force Institute of Technology, in May 1989.

Permanent Address: 632 Elmbank Street
Pittsburgh, PA 15226

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1990	Master's Thesis

4. TITLE AND SUBTITLE
DEVELOPMENT OF A HANDBOOK FOR CONFIGURATION MANAGERS WITH APPLICATIONS FOR THE PROGRAM/SYSTEM IN FULL-SCALE DEVELOPMENT

5. FUNDING NUMBERS

6. AUTHOR(S)
James E. Corbin, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GSM/LSY/90S-5

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This study resulted in the development of a Configuration Manager's Handbook that is intended to assist Air Force program offices and configuration management personnel apply the principles of configuration management to a product under development. The handbook can be used as a training document for in-coming personnel to a program office. It begins by briefly discussing the system acquisition life cycle as the domain in which a program is developed and the role of systems engineering in the development and design of the product. Configuration management is then introduced as a technical management control system that complements the technical actions undertaken during the systems engineering process. The handbook then proceeds to introduce, and in subsequent sections describe, the four processes that comprise configuration management: configuration identification, configuration audits, change management, and configuration status accounting. Finally, the uses of these processes for the program in full-scale development are discussed. This includes providing suggestions for the contents of Operating Instructions that should be produced by configuration managers to describe the specific applications of configuration management principles within the program office, and of Statement of Work tasks that should be required of the contractor. Keywords: [illegible]

14. SUBJECT TERMS
Configuration Management, Configuration, Configuration Handbook, Configuration Training, Air Force, [illegible]

15. NUMBER OF PAGES
350

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL